WL-TR-92-1054

(2)

EVOLUTIONARY ELECTRONIC COMBAT CONCEPTS (E2C2)

MERIT TECHNOLOGY, INC.
4035 COLONEL GLENN HIGHWAY
BEAVERCREEK, OHIO 45431

Report Date March 1992

Final Report for Period 15 March 1990 - 15 March 1992

Approved for public release; distribution is unlimited.

AVIONICS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6543

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related tnereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Michael Bohler

MICHAEL BOHLER
PROJECT ENGINEER

Phillip B. Hanselman Jr

PHILLIP B. HANSELMAN, CHIEF
TECHNOLOGY APPLICATIONS GROUP

Joseph C. Mower

JOSEPH C. MOWER,,Col, USAF
Deputy Chief
System Avionics Division
Avionics Directorate

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WL/AAAS, WPAFB, OH 45433-6543 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
16 Jun 92

3. REPORT TYPE AND DATES COVERED
Final 15 Mar 90 - 15 Mar 92

4. TITLE AND SUBTITLE
Evolutionary Electronic Combat Concepts (E2C2)

5. FUNDING NUMBERS
C- F33615-90-C-1403
PE - 63109F
PR - 3833
TA - 01
WU - 01

6. AUTHOR(S)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Merit Technology Inc.
4035 Colonel Glenn Highway
Beavercreek, Ohio 45431

8. PERFORMING ORGANIZATION REPORT NUMBER
4589-004

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Michael Bohler (513) 255-3765
Avionics Directorate (WL/AAAS)
Wright Laboratory
Wright-Patterson AFB OH 45433-6543

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
WL-TR-92-1054

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for Public Release;
Distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The Evolutionary Electronic Combat Concepts (E2C2) Program was an advanced research and development effort with objectives to design, develop, and demonstrate new and innovative techniques and concepts to aid the survivability of Air Force combat and combat support aircraft. The overall thrust of the effort was to demonstrate the importance of an integrated approach to offensive and defensive system management with the emphasis on electronic combat (EC) systems.

The techniques and concepts developed were used to provide the basis for an expert system offensive/defensive systems mediator capability for improved survivability, reduced threat exposure, elimination of conflicting EC actions, and reductions of aircraft systems energy emissions. This capability included the implementation of situation assessment algorithms and expert system decision strategies in a prototype Covert Integrated Combat Controller (CINTEC2).

CINTEC2 contains algorithms for managing offensive and defensive systems based upon threat and environmental situation assessment, mission parameters and status, and sensor/system conflict and data relationships. Included in CINTEC2 are algorithms

14. SUBJECT TERMS
Expert system, data fusion, sensor management

15. NUMBER OF PAGES
91

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std Z39-18
298-102

BLOCK 13 ABSTRACT CONTINUED

which perform search, track and identification of targets, fusion of sensor information, and expert system rules for decision making.

The major product of E2C2 effort was software that comprises CINTEC2. This included Ada software to perform target tracking and correlation, situation awareness, and a reconfigurable expert system knowledge base.

Preface

This Final Report was prepared for the United States Air Force, Aeronautical Systems Division, Wright Laboratories (WL), under contract number F33615-90-C-1403 by Merit Technology Incorporated. This document is the Final Report for the Evolutionary Electronic Combat Concepts (E2C2) Program sponsored by WL/AAAS-3 and AAWD. Mr. Michael Bohler was the government Program Manager for the contract. This Final Report documents all of the work performed on the E2C2 contract as outlined by the statement of work.

The E2C2 Program was an advanced research and development effort with objectives to design, develop, and demonstrate new and innovative techniques and concepts to aid the survivability of Air Force combat and combat support aircraft. The overall thrust of the effort was to demonstrate the importance of an integrated approach to offensive and defensive system management with the emphasis on electronic combat (EC) systems.

The techniques and concepts developed were used to provide the basis for an expert system offensive/defensive systems mediator capability for improved survivability, reduced *threat exposure, elimination of conflicting* EC actions, and reduction of aircraft systems energy emissions. This capability included the implementation of situation assessment algorithms and expert system decision strategies in a prototype Covert Integrated Electronic Combat Controller (CINTEC2).

CINTEC2 contains algorithms for managing offensive and defensive systems based upon threat and environmental situation assessment, mission parameters and status, and sensor/system conflict and data relationships. Included in CINTEC2 are algorithms which perform search, track, and identification of targets, fusion of sensor information, and expert system rules for decision making.

The major product of the E2C2 effort was software that comprises CINTEC2. This included Ada software to perform target tracking and correlation, situation awareness, and a reconfigurable expert system knowledge base based upon the MeriTool Inference Engine.

Simulation testing was performed in the Wright Laboratory Integrated Test Bed Facility using Merit's Cockpit Automation Technology Battle Area Tactical Simulation (CAT-BATS) to supply the aircraft model, avionics models, physical and threat environments. Aircraft, sensor, threat, and weapon system capabilities were representative of both present and future technology expectations for the next generation multi-role fighter aircraft.

Analysis of the CINTEC2 mission trials provide details on the ability of expert systems to provide decision making for the purposes of resource management, sensor allocation and control, and the benefits that can be derived from an integrated approach.

Finally, the benefits of the CINTEC2 system, as demonstrated by the mission trial testing, were identified. Benefits included:

(1) Automatic mediation of EC systems to reduce pilot workload while enhancing mission performance.
(2) Synergistic combination of sensor information and data for reduction of sensor energy emissions.
(3) Expert system alleviation of conflicting sensor actions through alternative decision strategies.
(4) Synthesized information capable of being directly assimilated by pilots.

These benefits, as well as lessons learned are documented in this final report.

Contents

Section **Page**

List of Figures

Acronyms

A3M	Air to Air Attack Management Program
ATB	Advanced Tactical Bomber
ATF	Advanced Tactical Fighter
ACM	Air Combat Maneuvering
ASPJ	Airborne Self Protection Jammer
BVR	Beyond Visual Range
CINTEC2	Covert Integrated Electronic Combat Controller
CM	Countermeasures
CPU	Central Processing Unit
CRT	Cathode Ray Tube
CSC	Computer Software Component
CSCI	Computer Software Configuration Item
DOF	Degrees of Freedom
E2C2	Evolutionary Electronic Combat Concepts Program
ECM	Electronic Countermeasures
EO	Electro-Optical
EOB	Electronic Order of Battle
ESM	Electronic Support Measures
EW	Electronic Warfare
EWWS	Electronic Warfare Warning System
FC	Fire Control
FEBA	Forward Edge of the Battle Area
FOR	Field of Regard
GCI	Ground Control Intercept
HDD	Head Down Display
HOSAT	Hands-on-Stick-and-Throttle
HUD	Head Up Display
ICNIA	Integrated Communications, Navigation, and Identification Avionics
ID	Identification
IFF	Identification Friend or Foe
INEWS	Integrated Electronic Warfare System
INF	Internetting Function
IR	Infrared

IRST	Infrared Search and Track
JTIDS	Joint Tactical Information Distribution System
LOS	Line-of-Sight
LPI	Low Probability of Intercept
MFD	Multifunction Display
MMR	Multi-Mode Radar
MRF	Multi-Role Fighter
MOE	Measure of Effectiveness
MSI	Multi-Source Integration
MTFM	MSI Track File Maintenance
PK	Probability of Kill
PVI	Pilot Vehicle Interface
RCS	Radar Cross Section
RF	Radio Frequency
RWR	Radar Warning Receiver
SAM	Surface to Air Missile
SGI	Silicon Graphics,Inc.
SNR	Signal to Noise Ratio
SOW	Statement of Work
SRS	Software Requirements Specification
SWAT	Subjective Workload Assessment Technique
TEAS	Threat Expert Analysis System
TFR	Terrain Following Radar
TSARS	Tactical Situation Assessment and Response Strategy
TWS	Track While Scan
USAF	United States Air Force
WL	Wright Laboratories
WVR	Within Visual Range

1.0 Introduction

This Final Report documents and summarizes the work accomplished by the Evolutionary Electronic Combat Concepts Program (E2C2) sponsored by the Wright Laboratories (WL) AAAS-3 and AAWD. The Dayton Division of Merit Technology, Inc. was the sole contractor for the E2C2 program.

This E2C2 Final Report covers the period of performance from contract award on 15 March 1990 through contract completion on 15 March 1992. It encompasses activities performed under the Statement of Work (SOW) from the E2C2 contract. The E2C2 effort was broken up into four separate phases as listed below.

Phase I:	System Requirements
Phase II:	Software Design
Phase III:	Software Implementation
Phase IV:	Software Test and Evaluation

1.1 E2C2 Program Overview

The objectives of the E2C2 Program were to design, develop, and demonstrate integrated offensive and defensive aircraft sensor and subsystems mediation and management through expert system control. The Covert Integrated Electronic Combat Controller (CINTEC2) utilizes available sensor information, changing mission objectives, ultimate mission goals, and sensor system relationships to provide improved sensor utilization, elimination of conflicting emissions, elimination of unnecessary emissions, improved survivability, and better situation awareness. A multisource integration data fusion capability was integrated to correlate data from onboard sensor systems and allow for operations in active or covert sensor modes.

The technology developed is applicable to such aircraft as the next generation Multirole Fighter (MRF) as well as future configurations of existing aircraft such as the F-16, F-15, ATF, or F117. The applicability of the technology to the different aircraft configurations is dependant more upon the future integrated avionics systems than upon the aircraft themselves. While the designs of the E2C2 effort were tailored using current aircraft sensor and avionics system capabilities, CINTEC2, to the fullest extent possible, takes advantage of functions of the integrated hardware and software envisioned for future use. The modular and flexible architecture of CINTEC2 is applicable to integration at either the signal or processor level and can be adapted for backfitting existing systems or applied to new systems. The technology can

be utilized for deconfliction of sensor requests from systems such as Quiet Knight, TSARS, or A^3M, or their associated rules can be incorporated into CINTEC2's decision strategy.

The future tactical combat environment will include enemy forces of increased numbers and technical sophistication. These sophisticated threat systems will be overlapping in coverage and operate across a broad spectral region. Each must be dealt with in order to achieve a successful mission. This environment requires a set of high performance detection and warning systems, coupled to an interactive response network which can provide timely and effective countermeasures against threat activity. Smart jammer systems, such as the ALQ-126B and ASPJ, developmental multi-spectral EW programs such as INEWS, and research programs such as the Threat Expert Analysis System (TEAS) and TSARS, are beginning to address the integration of defensive sensors and reactions systems to cope with this emerging threat environment.

The increasing sophistication and density of the threat environment has led to increasing emphasis on avoiding threat engagements. Accomplishing this objective requires improved offensive mission management systems which provide the ability to penetrate defended airspace without being engaged and, possibly, without being detected. Integrated offensive systems aimed at remaining hidden and minimizing exposure to the enemy weapon systems are being addressed in the stealth programs for both ATF and ATB. Other covert penetration programs, such as Quiet Knight, are being considered for special operations forces (SOF) missions as modifications of current generation aircraft. E2C2 addresses total avionics integration which includes the areas of offensive, defensive, navigation and communication avionics systems. However, the CINTEC2 prototype was restricted to considering offensive and defensive avionics systems.

The integration of these types of offensive and defensive systems for optimal defensive reaction and offensive mission achievement in the presence of the encountered threat is the primary objective of the E2C2 technology. Furthermore, the actions of eliminating interfering emissions, contradictory actions, and synergistic control of activities, responses, and mission objectives for this purpose cannot be achieved without an integrated approach. As we will show, integrated offensive and defensive responses are as important as improving the various avionics systems capabilities. Without this integration, the assessment and reactions proposed by the offensive and the defensive systems may be contradictory, further confusing an already complex environment. The result may be inaction or improper action, sacrificing stealth unnecessarily, catastrophically interfering with low altitude navigation, and minimizing the probability of a successful mission.

The CINTEC2 offensive/defensive mediation system algorithms are partitioned into two parts, situation awareness (SA module) and expert system planning (EW Planner module). Figure 1 represents the functional block diagram that depicts the major CINTEC2 functional elements. The Situation Awareness function takes care of the various external and internal inputs to the system. It handles tracking and correlation of sensor track file data from an external point of view, and mission considerations and aircraft system status from an internal point of view. This assessment is examined by CINTEC2's second functional element, the EW Planner. The EW Planner examines the data provided by the situation assessment function by use of an artificial intelligence expert system inference engine known as MeriTool. Depending upon the different possible assessment data elements, specific rules for the control, allocation, and use of sensors and systems will be executed and the corresponding actions performed in a non-conflicting manner.

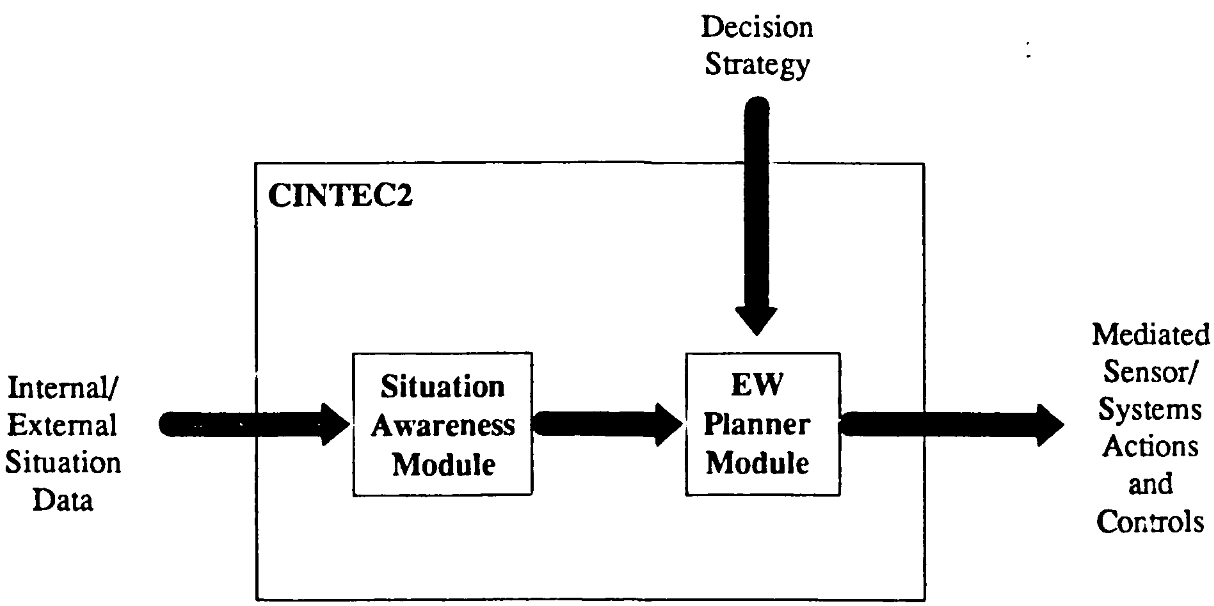


Figure 1 - CINTEC2 Top Level Functional Block Diagram

These functional components make up the nucleus of the CINTEC2 combat controller. In developing the software and algorithms that comprise them, work and research performed by other programs were incorporated where possible in an effort to save time and funding. Previous technical work that was incorporated in the E2C2 CINTEC2 approach includes:

1. Multisource Integration (MSI) - Algorithms for integrating sensor data to perform correlation and tracking of various targets.
2. MeriTool - An AI inference engine for forward and backward chaining through the use of textual rule sets and importable data elements.
3. Cockpit Automation Technology (CAT) - The Battle Area Tactical Simulation (BATS) was used to provide the aircraft, sensor, physical environment, and threat environment modeling necessary to drive the software.

Other programmatic efforts that are related to the E2C2 Program and have influenced the design and/or operation of the system include Integrated Communications, Navigation, and Identification Avionics (ICNIA), Air-to-Air Attack Management (A3M), Pilot's Associate, and ATACAP.

The E2C2 Program consisted of four phases. The first phase represented a study effort to analyze and determine the requirements of the overall mediation system. The second phase took the information and data gathered by the first phase and designed the software and data structures required to develop the prototype combat controller. The third phase implemented the software design from phase II in Ada and performed modular component testing. The fourth, and final phase performed integrated testing of the CINTEC2 combat controller system.

The basic program consisted of 24 months of technical effort which included the writing of a System Requirements Specification, Software Design Document, Interim Report, Software Test Plan, Software User's Manual, and this Final Report. No hardware development was required as part of the E2C2 contract.

The master schedule for the E2C2 Program is shown in Figure 2. All four phases are included in the master schedule. All contract deliverables are identified in the lower portion of the figure.

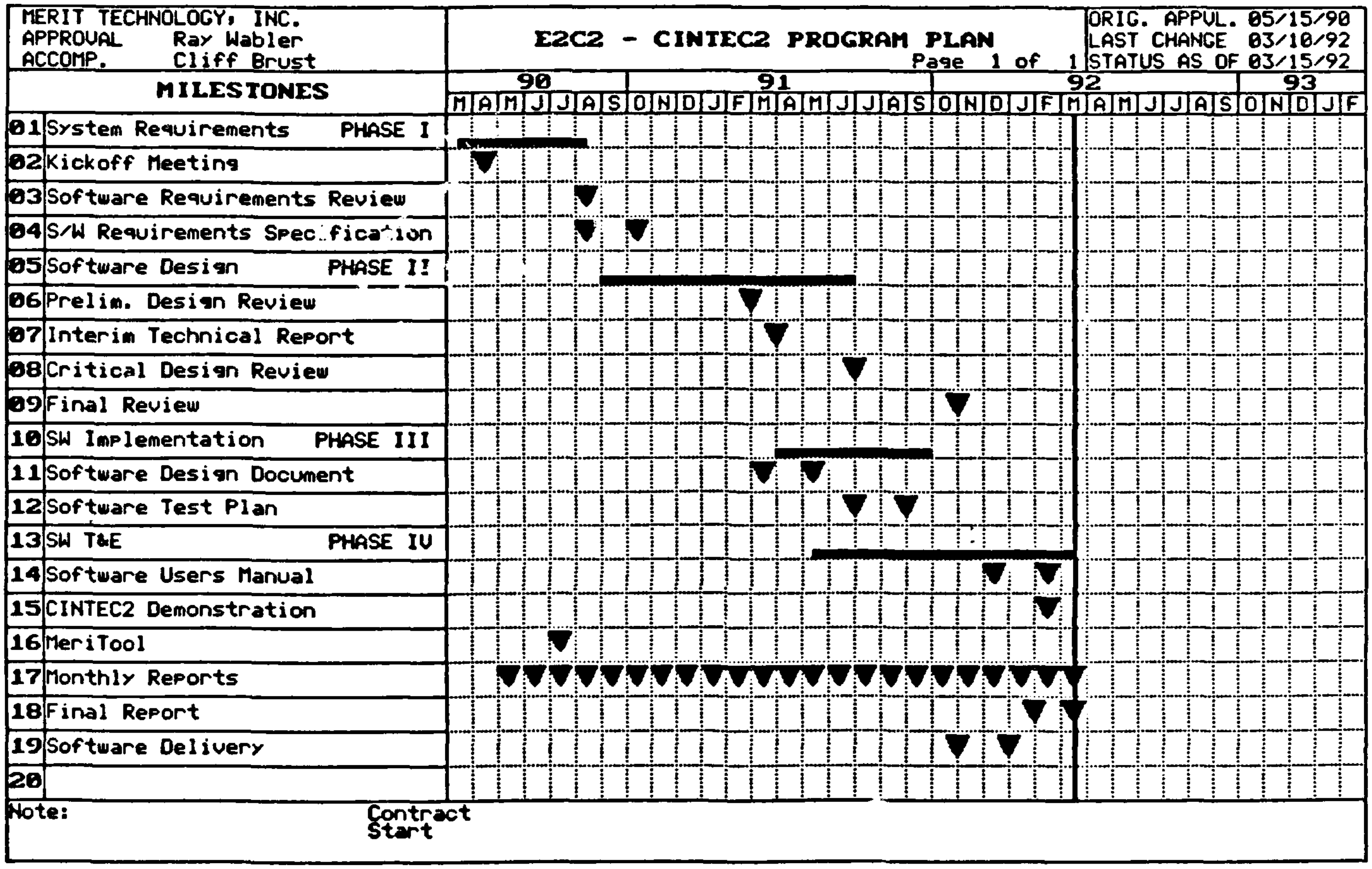


Figure 2 - Master Schedule

The CINTEC2 combat controller software is designed to assess the current environment, mission, and system status in order to provide for timely and non-conflicting sensor system actions. This software was developed in MIL-STD-1815A Ada for execution on a MicroVax computer system. The CINTEC2 software was tested by means of an external simulation environment known as CAT-BATS running on a Silicon Graphics computer system. Communications between CINTEC2 on the MicroVax and CAT-BATS on the Silicon Graphics was accomplished through an TCP/IP ethernet connection. Figure 3 represents the simulation development and test environment configured to test CINTEC2.

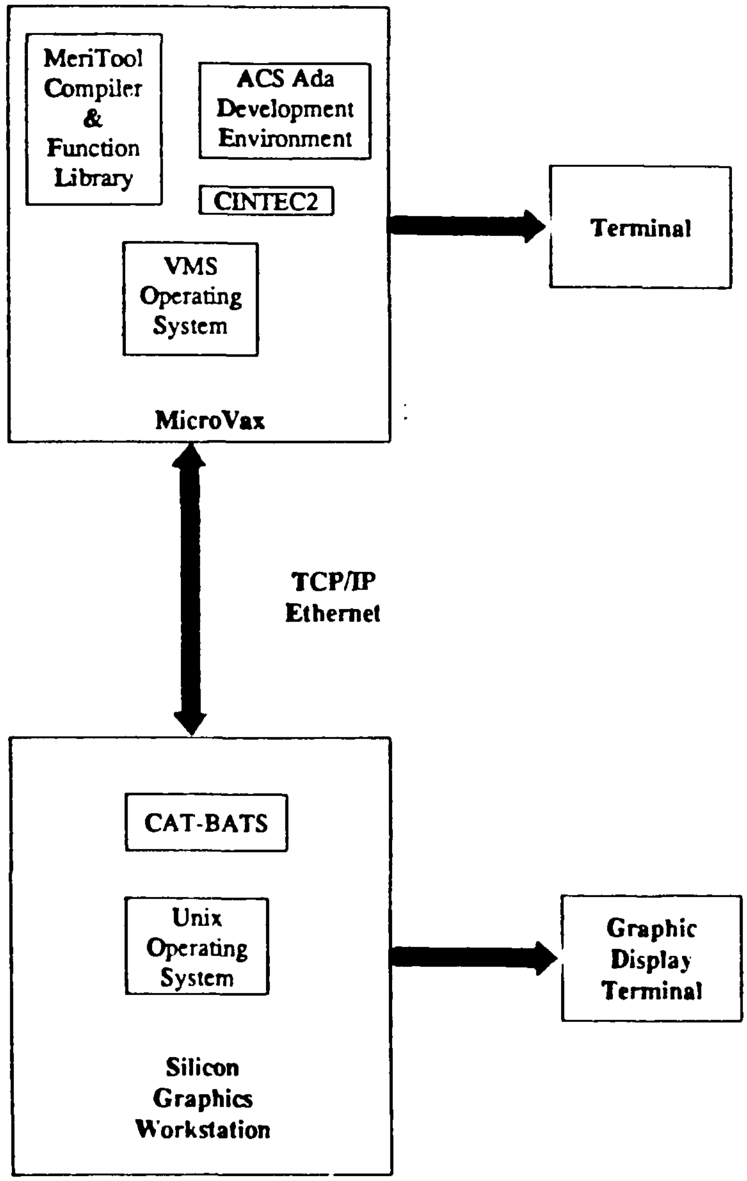


Figure 3 - Simulation Environment

1.1.2 Phase I Summary

Specific objectives associated with Phase I, the System Requirements Phase of the E2C2 Program were:

1. Research and analyze current and upcoming offensive and defensive avionics systems capabilities to determine the most appropriate systems for mediation.
2. Research and analyze current and forecast threat system capabilities to be used in decision strategy generation.
3. Establish valuable mission criteria to be used in decision strategy generation.

In performance of Phase I Merit drew upon many sources of information including inhouse experts as well as USAF civilian and military personnel in order to determine the most appropriate aircraft avionics systems for consideration as well as the most valuable criteria for establishing an expert system decision strategy for offensive/defensive systems integration, mediation and deconfliction. The E2C2 System Requirements Specification was delivered and a system requirements review was held at the end of this phase of the program.

1.1.3 Phase II Summary

Specific objectives associated with Phase II, the Software Design Phase of the E2C2 Program were:

1. Generate measures of effectiveness for evaluation of CINTEC2.
2. Establish basic mission scenario composition for CINTEC2 testing.
3. Perform the software design of the CINTEC2 combat controller.

In performance of Phase II, Merit developed the software architecture required for CINTEC2. In addition, the hardware requirements necessary to support the execution of CINTEC2 and the rest of the simulation components were identified. A wide variety of measures of effectiveness were identified during this phase of the program. The measures were narrowed down to those which would provide the most meaningful assessments of CINTEC2's capabilities. The mission scenario components were identified, the host airframe was ultimately selected as well as the finalization of the sensor suite. The E2C2 Software Design Document and Interim Report were delivered and a critical design review was held at the conclusion of Phase II.

1.1.4 Phase III Summary

Specific objectives associated with the Software Implementation Phase of E2C2 were:

1. Implement the CINTEC2 software design in Ada.
2. Perform individual software module component testing.
3. Generate the preliminary rule set decision strategy for the controller.
4. Modify the CAT-BATS simulation for integration with CINTEC2.
5. Develop simulation ethernet communications interface software.

In the performance of Phase III, Merit coded all of the software algorithms for the CINTEC2 controller. The software design, based upon object oriented techniques was fully implemented in Ada. Test driver programs used to test individual modules and module groupings were written in order to root out as many software bugs as possible before the final integration with the external CAT-BATS simulation. The preliminary rule set was generated based upon the mission, sensor, and environmental requirements identified in Phase I. The rule set was tested in a stand-alone mode of operation to help eliminate decision logic flaws. The E2C2 Software Test Plan was delivered at the end of Phase III and a preliminary CINTEC2 demonstration was presented.

1.1.5 Phase IV Summary

Specific objectives associated with the Test and Evaluation Phase of E2C2 were:

1. Integrated testing of CINTEC2 software components.
2. Final decision strategy (rule set) generation.
3. Test and Evaluation of the CINTEC2 decision strategy.
4. Evaluation of CINTEC2 against specific measures of effectiveness.

In the performance of Phase IV, Merit completed the software testing of CINTEC2 in an integrated environment consisting of the CINTEC2 controller and the CAT-BATS simulation. Refinements, modifications and additions to the preliminary decision strategy were made to eliminate any final bugs and elevate the robustness of the decision strategy. The final rule set and performance of the controller was then evaluated against specific measures of effectiveness. The end of Phase IV saw the delivery of the E2C2 CINTEC2 Software User's Manual and the E2C2 Final Report.

1.1.6 Final Report Overview

This Final Report documents the performance for Phase I through Phase IV of the E2C2 contract. In specific terms, Section 1.0 includes this overview and introduction. A system level description including a block diagram of CINTEC2 from a functional perspective are provided in the introductory portion of Section 2.0. These provide a means to identify the major functions and introduce the remaining subsections.

Subsections 2.1 through 2.3 address the major functions of CINTEC2 from a functional viewpoint. A block diagram of each function is provided and supplemented with detailed text descriptions of each block in the figure. The focus of the discussions in these subsections is on the processing and data flows within the system.

Section 3.0 documents software setups, testing, and evaluation used in determining CINTEC2 effectiveness. Simulation configuration is discussed along with mission scenario considerations and measures of effectiveness.

Finally, Section 4.0 summarizes the findings and discusses lessons learned over the course of the E2C2 contract relative to CINTEC2 and integrated offensive/defensive systems management. This includes areas of software development and test, mission simulation, and pilot interaction considerations. Further areas of research are also identified.

2.0 Functional Summary

The environment in which CINTEC2 operates is illustrated in Figure 4. It consists of simulated sensor systems (multi-mode radar, RWR, MWR, FLIR, optical, barrage jammer, chaff, flares, and launch warning detector), air to ground weapons (Maverick and Paveway II), threat environment, and aircraft dynamics. The complete simulation environment, with the exception of CINTEC2 processing, is provided by CAT-BATS. Mathematical modeling of the various sensor systems includes sensor envelope detection, line of sight, and object detection characteristic checking. The threat environment, including the effects of the barrage jammer is computed using signal level effects. The flight model is a full six degree-of-freedom (DOF) model.

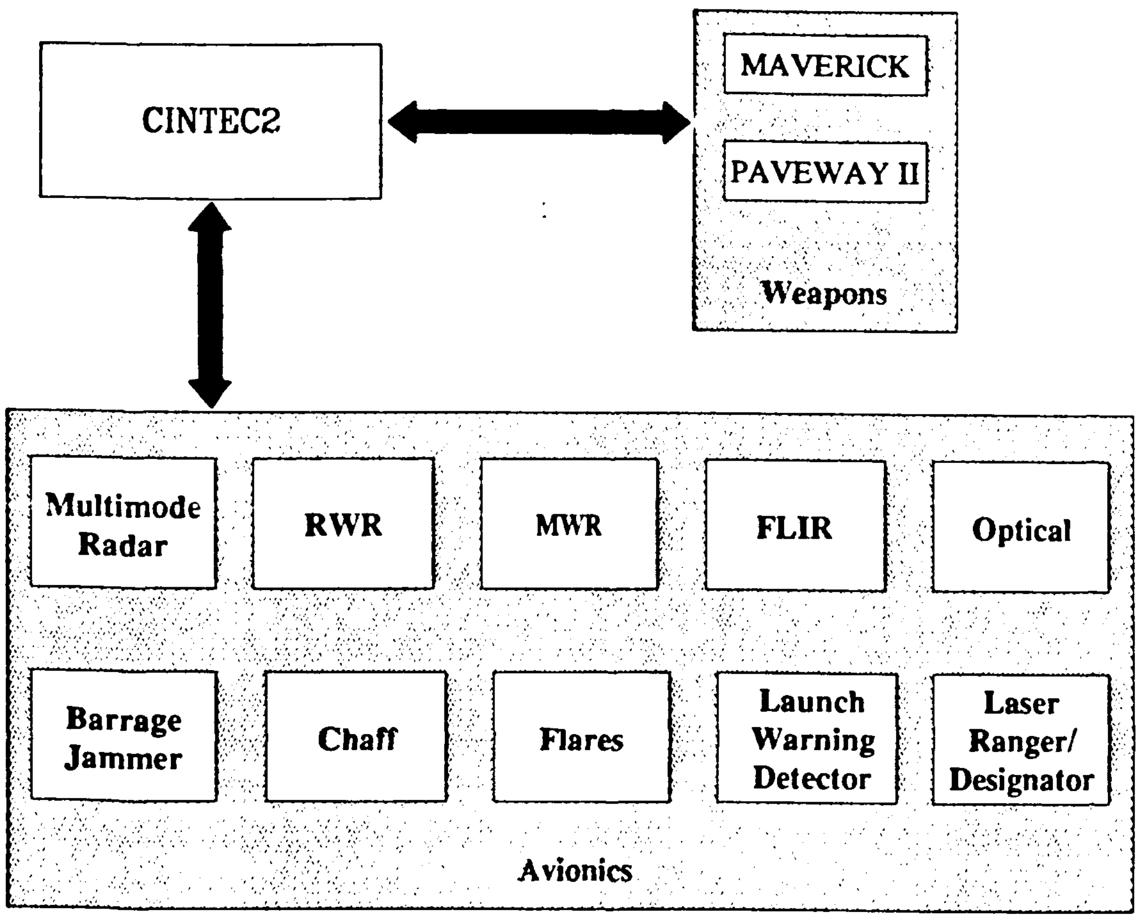


Figure 4 - CINTEC2 Avionics and Weapons

The CAT-BATS simulation environment runs on a Silicon Graphics Workstation under the UNIX operating system. communications to CINTEC2 is performed via a TCP/IP ethernet link in a completely synchronous fashion. All simulations were run in simulated real time since real-time operation of the controller at this stage of the E2C2 program was not necessary. A MicroVax III computer system with Ada and Fusion TCP communications software was the target processor for CINTEC2. The CINTEC2 controller software is a deliverable under the E2C2 contract, but the CAT-BATS simulation environment is not. A 6-month license for CAT-BATS was provided at the end of contract so that further CINTEC2 testing could be conducted.

Figure 5 provides a functional block diagram illustrating the various application functions of CINTEC2. Interfaces between the various components including the sensor systems is shown in the figure. CINTEC2 is shown to be composed of the three major function groups and the associated data structures.

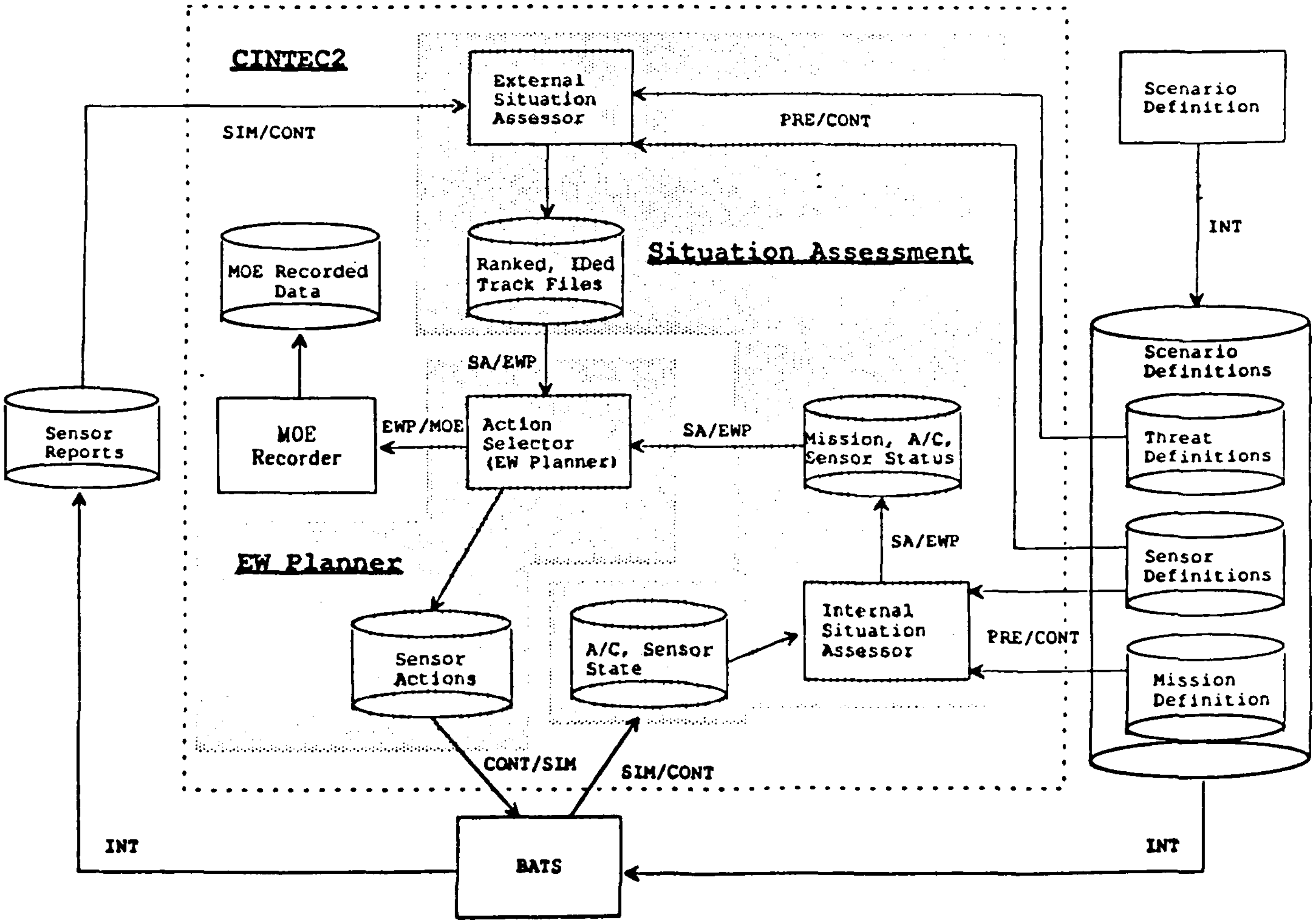


Figure 5 - CINTEC2 Functional Block Diagram

2.1 CINTEC2 Driver

The execution and control of the CINTEC2 Situation Awareness and EW Planning software is performed by the CINTEC2 Driver Software. The CINTEC2 Driver consists of the overall CINTEC2 function management and operating system interfaces required to execute. CINTEC2 Driver function management consists of performing the correct sequence of operations for the gathering of environmental and system status information and triggering the desired subfunctions at the correct time.

The CINTEC2 Driver was developed for operation on a MicroVax III target processor. It serves as the interface between the operator as well as the external simulation environment. The driver performs the functions associated initializing the various CINTEC2 data structures and receiving and transmitting data with the external CAT-BATS simulation software. It also performs the function of packing and unpacking the differing data formats between the simulation hardware components. The driver is also responsible for providing the incoming simulation data to the Situation Assessment and EW Planner functions as well as initiating these functions. After CINTEC2 processing has completed, it must also handle the generated actions and commands in terms of passing them on to the external simulation for execution in the various models. Figure 6 depicts the functional flow of the CINTEC2 Driver function.

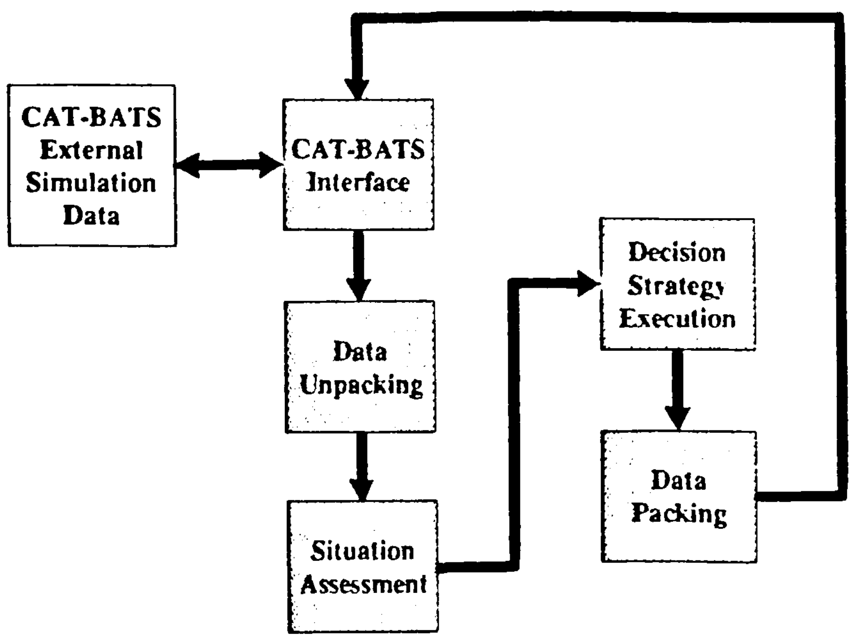


Figure 6 - CINTEC2 Driver Functional Block Diagram

2.2 Situation Assessment Function

The Situation Assessment (SA) function serves to accumulate and organize the information from the external simulation environment and pass the fused and correlated information on to the EW Planner sensor mediation decision strategy. This function handles all situation information including sensor status, mission status, mission objectives, weaponeering, and all sensor track file information. In order to accomplish its function it acquires its data from the CINTEC2 Driver and places the various data components into object oriented data structures.

The SA function is divided into two segments within CINTEC2. These two divisions consist of the external SA and the internal SA functions as shown in Figure 7. The external SA function monitors the physical environment, mission environment, and the threat environment. The internal SA function monitors the internal aircraft functions such as sensor systems and avionics. Together, a picture of the circumstances under which decisions for sensor and systems control is performed is maintained and continuously updated. The Situation Assessor data structures that track the various information is depicted in Figure 8.

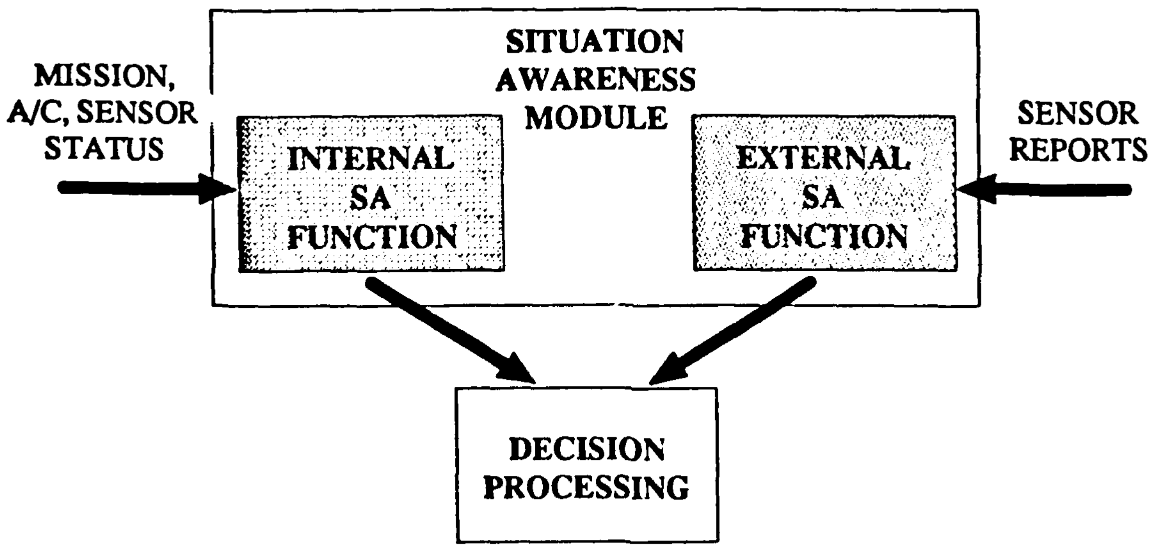


Figure 7 - Situation Assessment Functional Block Diagram

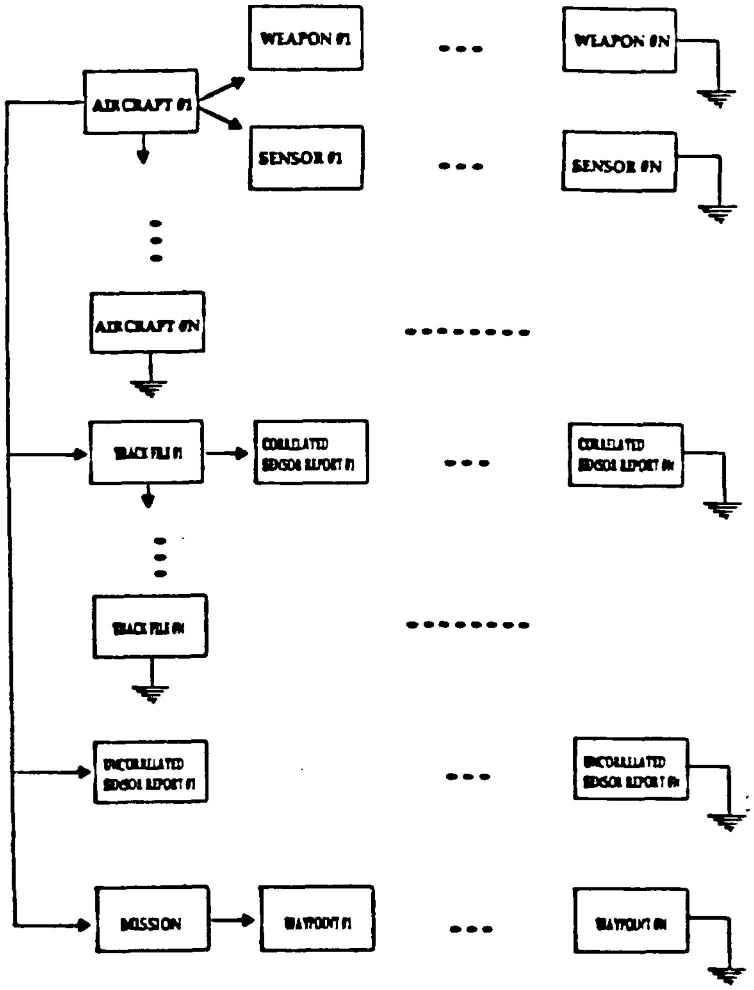


Figure 8 - Situation Assessor Data Structure

The CINTEC2 SA function is coded completely in Ada. That is, there is no artificial intelligence decision making being performed within this particular function of CINTEC2. The external SA function was based upon software developed for the Multi-Source Integration (MSI) modeling project and was originally written in FORTRAN. The algorithms were partially re-designed to take advantage of object oriented data handling techniques and common subroutine interface standards that could be employed with the use of Ada. The software was then coded and tested in a stand-alone testing environment. Testing was performed on individual modules as well as module sets to determine proper operation of the software.

2.2.1 External Situation Assessment

The External Situation function assesses that component of the situation which is external to the ownship aircraft. Thus, it includes information about threats, sensor detections, enemy weapons and sensors, hostile intent, and other objects plans, goals, and actions. There are three functions performed to assess the external situation:

(1) Data Fusion
(2) Track Interpretation (Identification and Track Correlation)
(3) Threat Ranking (designed but not implemented)

Basic information flow and functional composition of the External Situation Assessor is shown in Figure 9.

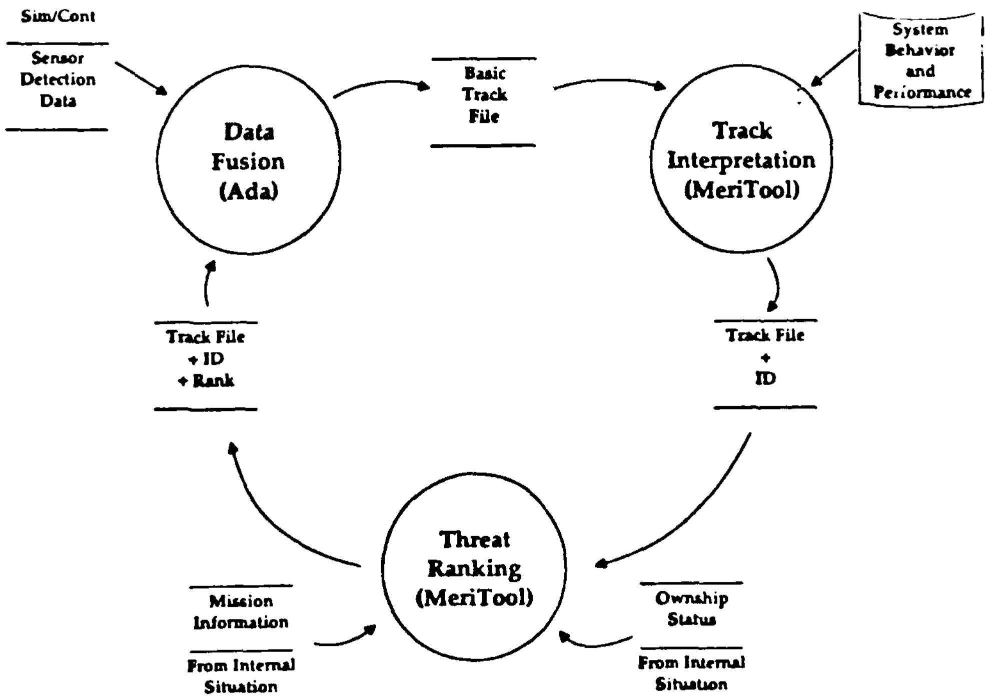


Figure 9 - External SA Functional Block Diagram

The External Situation Assessment (SA) Function provides the sensor track file algorithms necessary to perform multi-source integration and tracking of targets. The SA function performs its duties through a hierarchy of track file data. At the lowest level of the hierarchy are the Sensor Track Files (STF). The STF are unique to a given sensor system and represent the individual capabilities associated with an individual sensor system. The MultiSource Integration (MSI) track files are formed by fusing the various STFs that have been reported. Figure 10 represents the track file structure and hierarchy.

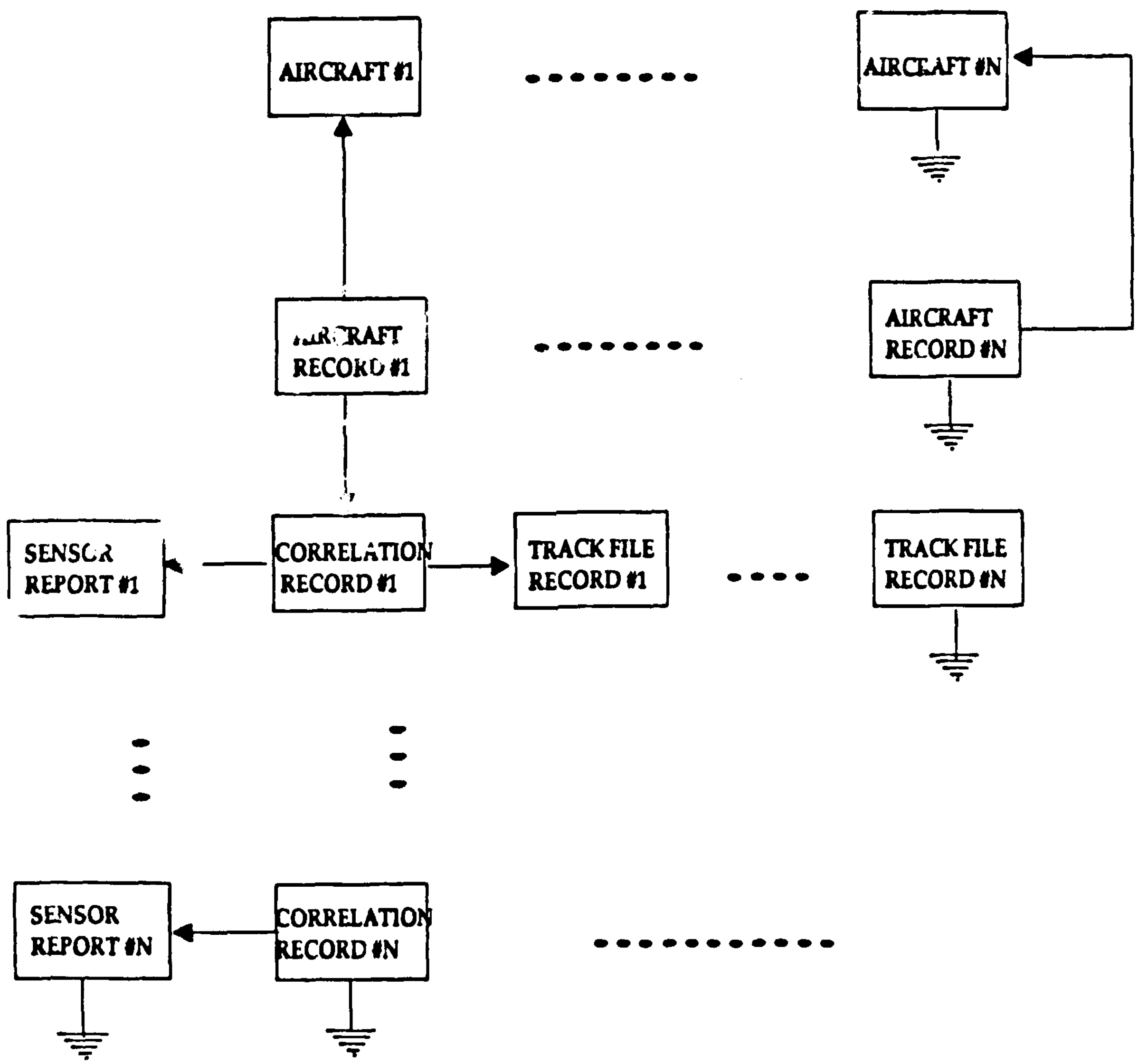


Figure 10 - Track File Hierarchy

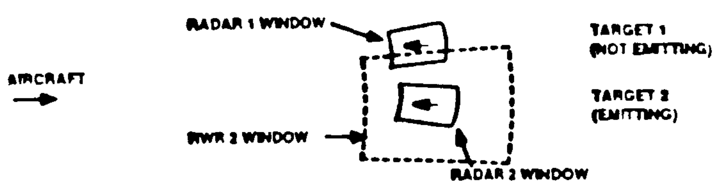
2.2.1.1 Data Fusion and Track Files (Sensor/Track File Correlation)

Data fusion in the external situation assessor is the function responsible for collecting data from the sensor suite and organizing it as a collection of object tracks. Inputs to the data fusion function come from two basic sources: previous fusion efforts (in the form of existing track files) and newly acquired sensor reports. After the first sensor detection cycle, the CINTEC2 Controller will have an existing track file which will also be available to the external situation data fusion function.

The basic processing loop executes for each sensor report. A subloop considers each existing track, testing the sensor report against the track. Where the report is both geographically feasible and feasible in characteristics, the report is added to or associated with the track. Notice that this may result in a single report being added to more than one track.

"Geographic feasibility" as used here is a window-based determination. Each sensor has an associated error or "slop" factor. That error factor is used in combination with the position contained in the sensor report to generate a polygon or "window" representing the likely true position of the reported object. A similar window is generated for the track being considered. If the windows overlap, the sensor report is considered geographically feasible for that track.

The sensor track correlator is designed to take sensor reports and merge them together into observation files for access by the decision strategy rule set. Primary sensor (sensors which have the capability to track targets such as radar, JTIDS, IRST) and secondary sensor (non-tracking sensors such as RWR, launch warning detector) tracks are combined and correlated to one another through a spatial windowing technique sometimes referred to as the "nearest neighbor algorithm". Figure 11 shows an example of how the nearest neighbor algorithm is used to correlate an RWR and radar report into a correlated track file record.



TRACK FILE 1	RADAR 1 REPORT	RWR 2 REPORT
TRACK FILE 2	RADAR 2 REPORT	RWR 2 REPORT

Figure 11 - Nearest Neighbor Geographic Feasibility Correlation

Each sensor has a correlation window which equates to its own specified accuracy. When a given track is reported to the situation assessor, the tracking sensor's window is compared to the MSI track file window which is based upon the MSI track files initializing sensors correlation window for the given track being compared. If the windows overlap that sensor report is considered to be correlated with the track file. This functionality, although implemented in the CINTEC2 scheme, assumes that all sensor correlation windows are identical. Figure 12 depicts the flow of the correlation algorithm.

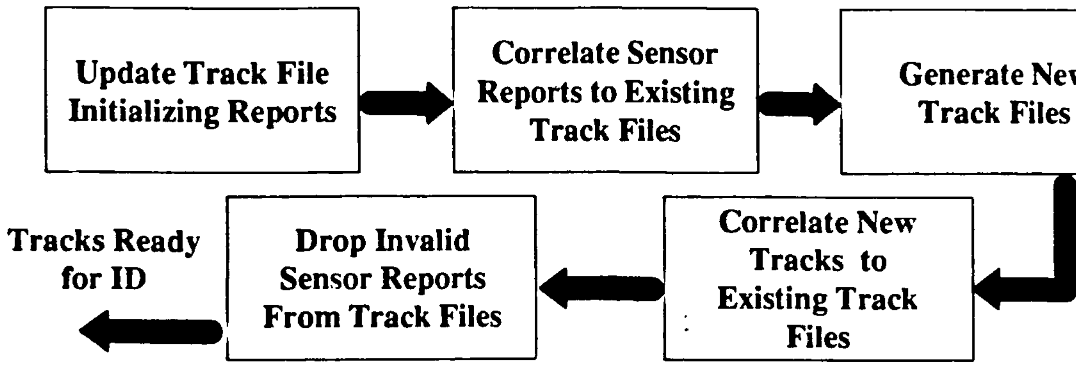
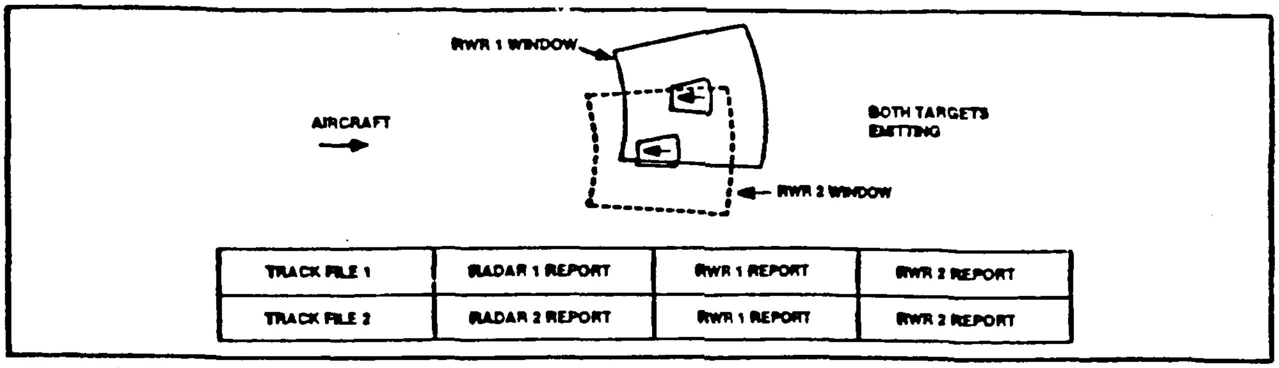
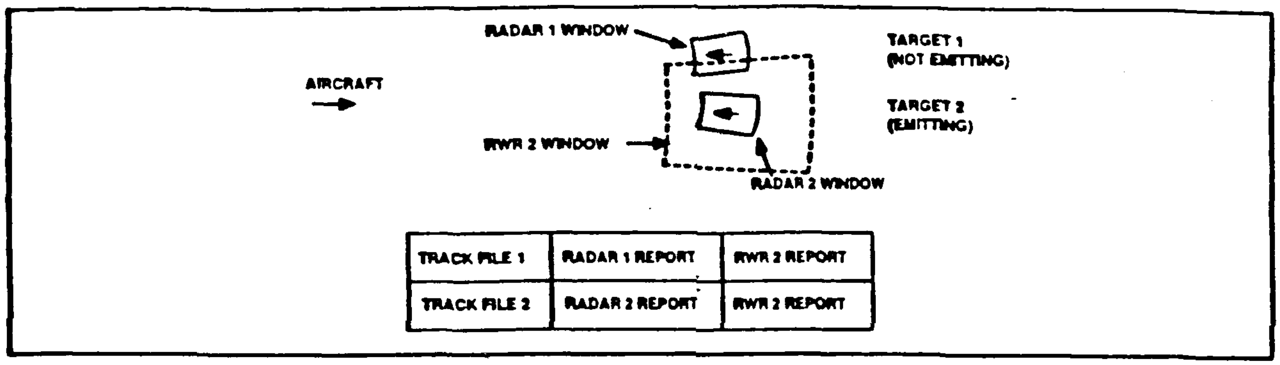


Figure 12 - Sensor Report Correlation Algorithm

The sensor report to track file correlation algorithm represents the nucleus of the External Situation Assessment Function. Each sensor report is compared to each track file to see if there is correlation. A sensor report's azimuth, range, and elevation are compared to that of the track files initializing sensor report. If the correlation windows overlap in three dimensions the report is "correlated" to the track file. If the report is an update to an old sensor report already in the track file, that report is replaced with the new one. If a report did not correlate to an existing track file, a new track is initiated. Note that due to window overlap a single sensor report can correlate to multiple track files. Figure 13 shows how sensor reports can correlate to more than one track file.



MULTIPLE CORRELATION - TWO SENSOR REPORTS (FROM SAME SENSOR) TO ONE TRACK FILE



MULTIPLE CORRELATION - ONE SENSOR REPORT TO TWO TRACK FILES

Figure 13 - Sensor Report Multiple Correlation

"Characteristic feasibility" is a test of non-position information, typically identification. Identification information may be pictured as a tangled hierarchy or network, with less precise identifications dominating more precise ones. Thus, an identification of "fighter" is less precise than that of "F-15" and so would dominate. A report is feasible in characteristics to a track if there is no identification information in the report, no identification information in the track, or if the identification information in the report and the track are related to each other through the dominance relation described informally above. If a sensor report cannot be associated with any existing track, then a new track is created, and the sensor report is associated with the new track.

After all sensor reports have been processed, a second processing loop is executed over those tracks which have not had any sensor reports associated with them in this cycle. For each of those tracks, the most recent update time is compared to the current time less a decay constant (which may vary based on sensor type). If a track has not been updated by a sensor report within the specified time, it is dropped from the track file. If a track has been updated within the limit, its position is dead-reckoned from the existing track position information. Tracks with a single position point are presumed stationary. Dead-reckoned tracks are given sensor type "INFERRED".

The output of the data fusion function is a track file, which is a collection of tracks. Each track has a unique track identifier (ID), and includes a time-sorted list of positions. A track file may (but need not) also include an assigned identification and a threat ranking. These fields are typically filled and used by later processing stages.

Track interpretation (Figure 14) is the process of associating a real-world object (e.g. aircraft, ground site, etc.) with a track. Many times, the association will be between a track and a set or class of objects, such as all surface-to-air missile launchers of a particular type (e.g. SA-6). Using the object hierarchy mentioned earlier in characteristics matching, track interpretations may be refined by moving through the hierarchy to more precisely defined objects (e.g. SA-6 site).

Inputs to interpretation are background knowledge concerning the objects of interest (e.g. aircraft and weapon performance) and the tracks resulting from the data fusion function. The track file produced by the external data fusion function is an input to the external track interpretation function. Track files produced by this function are also acceptable inputs.

Descriptions of the performance and behavior of all objects which can be tracked are available as an input to the track interpretation function. This information includes:

- Emitter type listings by platform
- Platform movement parameters (speeds, maneuverability, etc.)
- Object composition information (e.g. aircraft carry missiles -- therefore an aircraft object can appear to divide into multiple objects)
- Platform mission objectives

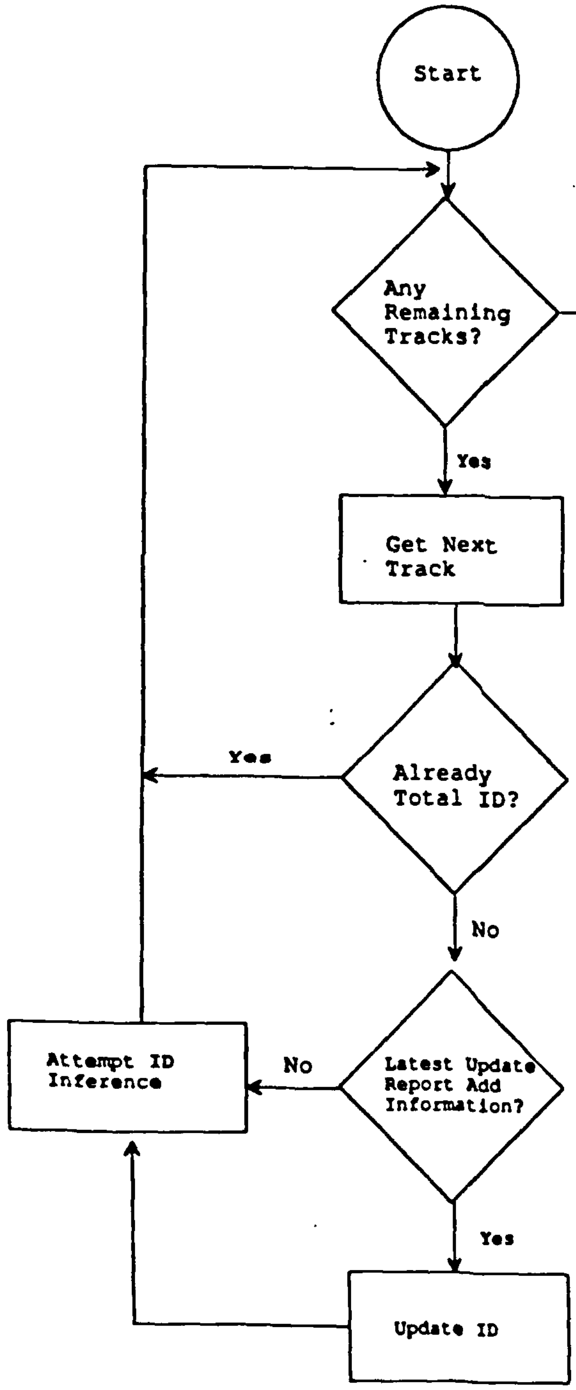


Figure 14 - Track Interpretation Processing Flow Diagram

Track interpretation considers each track in turn. If the track has a total identification associated with it, interpretation processing stops. "Total identification" means that the identification associated with the track is at a base level in the object hierarchy and cannot be further decomposed. For example, a particular search radar associated with a specific, known SAM site would be at base level.

If a track has not been totally identified, any sensor reports associated with the track in the current time period are considered to see if they provide additional information for identification. Such additional information might include location information or emission information. If such additional information is present in the current reports, the track identification is updated appropriately.

Finally, whether or not the current sensor reports contain explicit additional identification information, an inference process is performed on the entire track file, matching the recorded behavior of the track object with the behaviors and performances available to the system. If there is a match, the track is identified as of the type which would evidence such behavior. Thus, for example, if a particular cycling of sensor modes is detected, a site may be identified as being of some known type.

2.2.1.2 Threat Assessment

Threat assessment for identification purposes was not implemented as a part of the overall CINTEC2 system even though it was designed. The reason for this was that it was determined to be of secondary importance relative to the tasks of performing offensive/defensive sensor system mediation and deconfliction. The threat assessment function could offer some very valuable inputs to the overall integration of avionics systems in the future, however it does not add significantly to the research and development of the prototype CINTEC2 system at this stage. The reason for this is that other inputs to the CINTEC2 system such as mission information (briefed threat laydown, or EOB) can exercise the decision making rule set in a similar fashion to the actual real-time identification of threat systems and the incorporation of this information into the decision making process. The operation of the threat assessment function is presented here for completeness of the final report.

The output of the track interpretation function is a track file containing tracks, each of which has a field which holds platform identification information. Platform identification information includes platform type, unique identifier (where available), hostile/neutral/friendly,

and other similar information, and references the object hierarchy. Tracks produced by the track interpretation function are interchangeable with tracks produced by data fusion or threat ranking in any function which takes a track or multiple tracks as input.

Threat ranking assigns a priority to tracks based on their time-weighted potential for preventing successful execution of the current mission. Tentatively, two threats which pose the same potential are ordered by closeness in time to the time of evaluation, with the closer ranked more threatening. The precise tradeoff between potential and timing is determined by user selection. Destruction of the ownship is considered to be less than successful execution. The threat ranking criteria are subject to change and refinement during detailed design. Only those tracks which have been interpreted as hostile or potentially hostile are ranked by the threat ranking process, and the output threat ranking is totally ordered.

As with track interpretation, threat ranking depends heavily on the existence of a track file. Ranking also relies on background information about capabilities to assess the threat posed by each track.

Information describing the ownship mission, including but not limited to waypoints, target data (including location, type, and time on target (TOT)), weapon loadout, weapon release points, flight co-members, communications plan, and command structure, are input to the threat ranking function.

Threat ranking rules are inference rules which associate threat scores with indicators such as weapon type, position, sensor complement, alignment, and behavior.

Threat ranking has two component subfunctions:

(a) determine_threat

Calculates a raw threat score for an input track. This threat score is 0 for friendly and neutral tracks, and is a measure of potential for mission disruption for hostile tracks. Threat measures consider, at the least, probability of kill and closeness of threat in time. The threat score for a track is determined by the threat ranking rules.

(b) order_threats

Orders all hostile tracks based on their raw threat score. Ties are broken using criteria to be determined during design. The function order_threats then assigns a threat ranking to the ordered threats, ranging from 1 through n, where n is the number of hostile tracks.

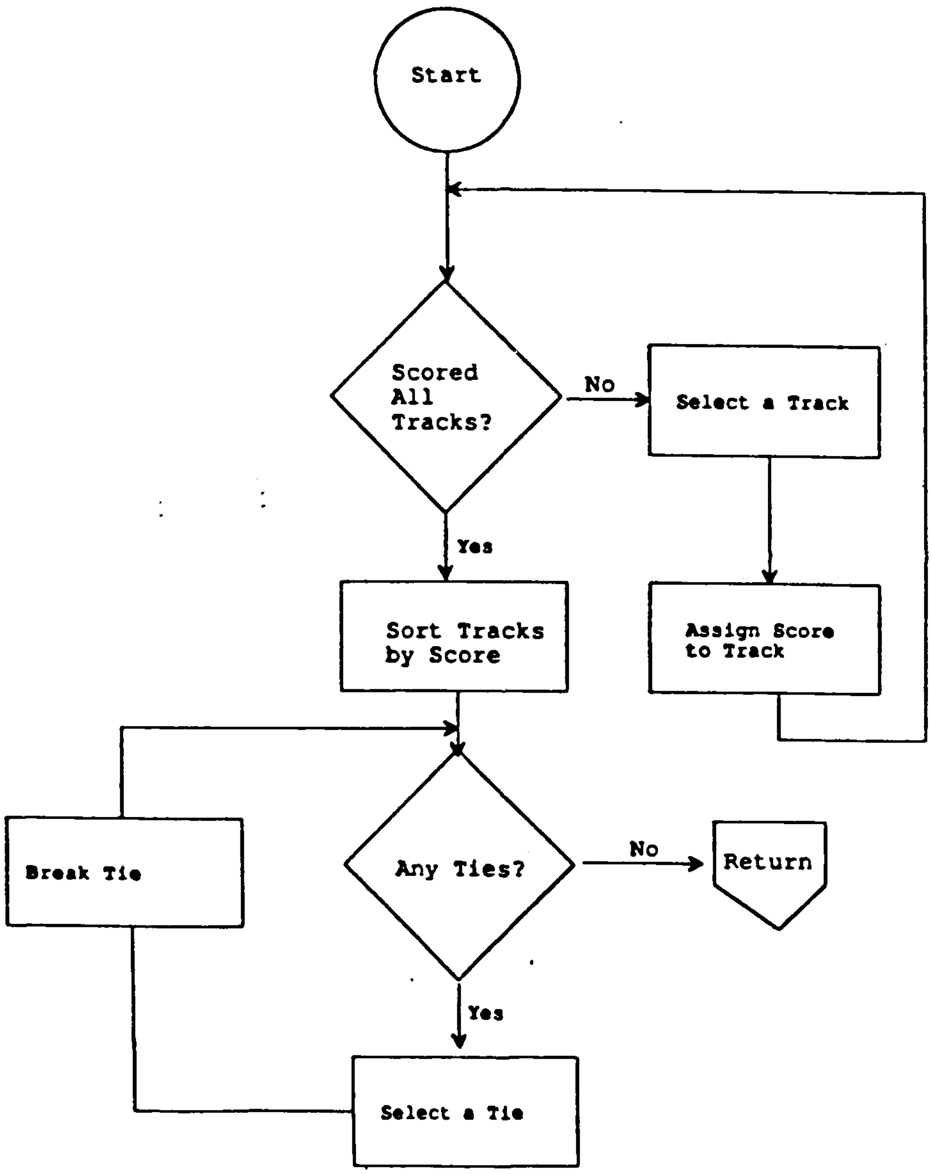


Figure 15 - Threat Ranking Processing Flow

The output of the threat ranking function is a track file containing ranked tracks, each of which has a field which holds threat ranking information. All rankings are non-negative integers, and are contiguous. The rank of any track evaluated as friendly or neutral is 0. With the exception of 0 rankings, every ranking is assigned to only one track. The least threatening track (smallest potential disruption of mission) is ranked 1, with all other hostile tracks assigned greater, unique rankings. The most threatening track (greatest potential disruption of mission) is assigned an integer ranking greater than the ranking of any other track.

Tracks produced by the threat ranking function are interchangeable with tracks produced by data fusion or track interpretation in any function which takes a track or multiple tracks as input.

2.2.2 Internal Situation Assessment

The internal situation is that which describes the aircraft and its subsystems, without regard to external platforms. Much of the requisite data which is input to the internal situation function is taken directly from the flight simulation, rather than being acquired through simulated sensors. Most internal situation processing is data format conversion to make these types of information more readily available to the CINTEC2 Controller.

2.2.2.1 Ownship Status and Sensor Assessment

Aircraft status in this context concerns the physical state of the aircraft -- its position and path. The aircraft status function retrieves the aircraft status inputs from the flight model current state, reformats the data as MeriTool assertions using MeriTool object definitions which were created during design, and asserts the data into the MeriTool working memory. The outputs of the aircraft status function are CINTEC2 compatible data representations of the input data, in the form of MeriTool assertions.

Sensor status is the current configuration of the sensor suite, including the mode and option settings on all sensors. Detection data produced by the sensors is input to the data fusion function of the External Situation Assessor. The inputs to the sensor status function are the current mode and option settings for each of the sensors. Typical modes include (but are not limited to): ON, OFF, STANDBY, and INACTIVE. Settings include information like current range selected on multi-mode radar, and radar mode (track-while-scan, single-target-track, etc.).

The sensor status function retrieves the sensor status inputs from the aircraft model current state, reformats the data as MeriTool assertions using MeriTool object definitions, and asserts the data into the MeriTool working memory. The outputs of the sensor status function are CINTEC2 compatible data representations of the input data, in the form of MeriTool assertions.

2.2.2.2 Mission Assessment

Mission status is an assessment of the current state of the assigned mission. Specific assessments were defined during design, and include constructs which capture the sense of DELAYED, ON-SCHEDULE, LEG-ABANDONED, THREATENED, and ABORTED. The inputs to mission status are the assigned mission and the ownship performance to this point. The assigned mission input has the form of a time-ordered collection of waypoints, each of which specifies a time and three-dimensional location. Waypoints are treated as 4D position objectives, and it is assumed that closeness of fit of the actual flight path with the defined waypoints provides an accurate indicator of mission status. Each waypoint also has an assigned type, which is one of NAVIGAT[illegible]AL or WEAPON_RELEASE. The type of a waypoint can be used as an indicator of th[illegible]tion to be performed at or near that waypoint, and success or failure to execute the related action is also used as an indicator of mission status.

Ownship performance is a track file whose ID is ownship, with track points annotated with any actions which were performed at that point (such as weapon release). Typically, a new track point is added to the ownship basic track file on each simulation cycle. The outputs of the mission status function are CINTEC2 compatible data representations of the derived mission status information, in the form of MeriTool assertions.

2.3 EW Planner Function

The EW Planner Function automatically commands all of the controllable sensors in the E2C2 sensor suite. The sensors and/or systems available for control are as follows:

1. Barrage Jammer
2. Multimode Radar (MMR)
3. Radar Warning Receiver (RWR)
4. Missile Warning Receiver (MWR)
5. Launch Warning Detector (LWD)
6. Forward Looking Infrared (FLIR)
7. Maverick Missile Optical Sensor (prior to flyout)

In addition to control of this sensor suite, the planner function can also make recommendations to pop chaff or flares, perform high G maneuvers, and cue specific sensor systems for upcoming events. These recommendations are based upon the current situation assessment and mission objectives. The performance of these specific functions lies within the given programmed decision strategy of the EW Planner Function.

The EW Planner performs its task on each and every simulation step. That is, it is run synchronously with the external simulation. For every iteration, all of the various status and information representations are examined by the MeriTool inference engine. This causes specific rule "firings" based upon the data. These rule firings are then, in turn, translated into commands for the sensor and system models running as part of the external simulation. They are finally passed to the simulation for execution. The EW Planner is partitioned into three component functions listed below. Figure 16 diagrams their relationship to one another.

1. Generate Sensor Actions
2. Detect Action Conflicts
3. Resolve Action Conflicts

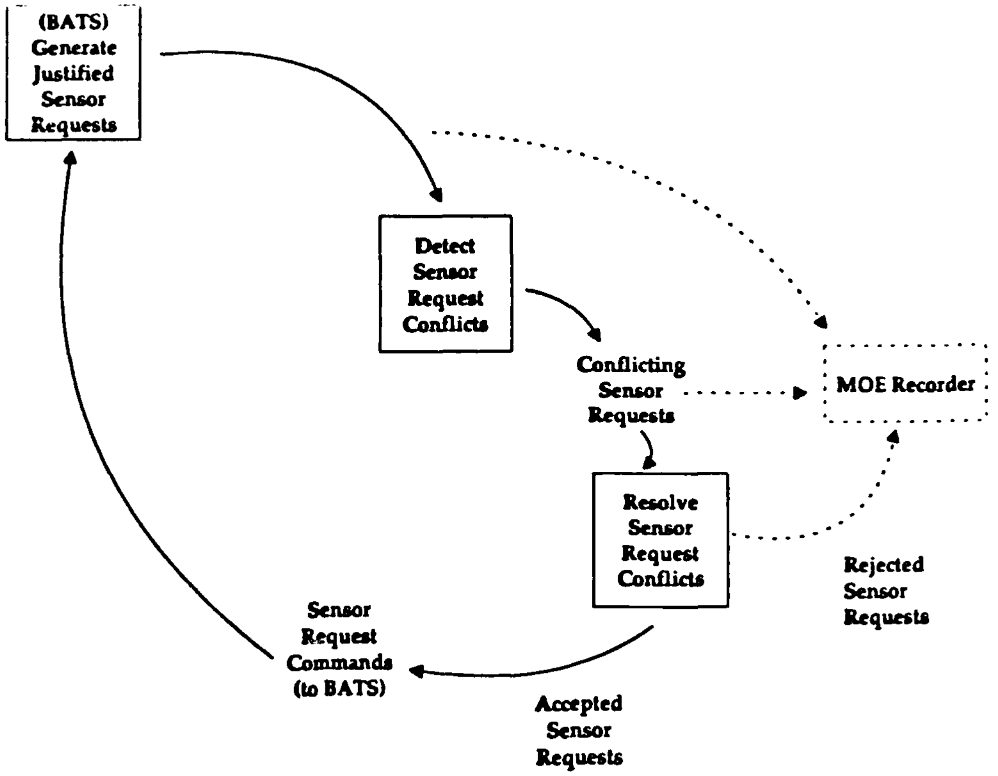


Figure 16 - EW Planner Functional Block Diagram

2.3.1 Decision Strategy

All possible sensor actions are considered to produce a filtered list of actions which can be justified in the current situation. This list of justified actions is then examined to find actions which conflict with either other actions or with overall mission goals and plans. When conflicts are detected, they are resolved by deleting one or more actions. Non-conflicting actions and those conflicting actions which survived the conflict resolution process are passed to the simulation for execution.

A sensor action is "justified" if it is both physically possible to perform in the current environment and performing it would enable an activity which supports some goal of the pilot. For example, turning on the radar can support an information-gathering goal, but cannot (trivially) be performed if the radar is already on. This component uses an "overgeneration" strategy to produce all possible sensor actions for the current situation.

Part of the rule set has conditions of the rules constructed so that the rule themselves are specialized to perform tests of the internal and external situation (as produced by the situation assessor) and the conclusions are sensor actions. This rule set is part of the initial system load, and represents the "justification" of sensor actions. Conditions of the rules are tested against the data provided by the situation assessment. This processing is provided by the inference engine. Rules whose conditions are satisfied will produce sensor action recommendations.

The output of the Justified Sensor Action Generation process is a collection of sensor actions. Every action in the output collection is both physically possible and supports some goal of the pilot. It is more than likely that many of the generated actions will conflict.

Intra-sensor conflicts are present when two actions seek to place a single sensor into mutually exclusive states. Inter-sensor conflicts are typically emission-detection conflicts, though in more sophisticated systems, such conflicts could arise from attempts to share apertures or processing suites. Sensor-mission conflicts indicate a violation of some mission goal, such as covertness, by the designated sensor action. This component detects such conflicts among the sensor actions generated.

A third input to conflict detection is a MeriTool rule set containing rules whose conditions are sensor actions and whose conclusions are conflict descriptions.

2.3.1.1 Rule Set

The rule set contains the actual inference rules that are being executed in order to perform a given decision strategy. The rule set itself is divided into five sections, these sections each perform functions relating to a given CINTEC2 function but also have interdependencies with each other. The five rule set divisions are as follows:

(1) Sensor and Subsystem Rules
(2) Mission Rules
(3) Threat Rules
(4) Weapon Rules
(5) Environmental Rules

These divisions each contain a variety of rules for analyzing the situation awareness data and permitting or denying actions based upon the conglomerated circumstances. Each rule set division can influence or effect the others. As an example, a given mission phase (Mission Rule Set Division) which requires a target to be acquired may require the use of an active emitting sensor in order to accomplish the task. If, however, the given sensor is currently occupied or malfunctioning (Sensor and Subsystem Rule Set), this action would be denied, and an alternative strategy would have to then be implemented in order to accomplish the goal.

The final rule set that was used in testing the CINTEC2 combat controller is included in Appendix B. This rule set is written in the MeriTool rule set format which is documented in the MeriTool 2.2 Users Manual.

3.0 Testing and Evaluation

This section summarizes the E2C2 simulation used for testing of the CINTEC2 combat controller software, the test program that was conducted and the results obtained from the testing.

3.1 Simulation Configuration

The simulation used for the testing of CINTEC2 consisted of the CAT-BATS aircraft, avionics, and threat environment simulation modeling software and the CINTEC2 combat controller software. This section contains an overview of the simulation configuration for E2C2. The hardware platforms on which the CINTEC2 and CAT-BATS software were executed were not provided as part of the E2C2 program, but consisted of existing Integrated Test Bed laboratory computer systems. The simulation configuration is depicted below in Figure 17.

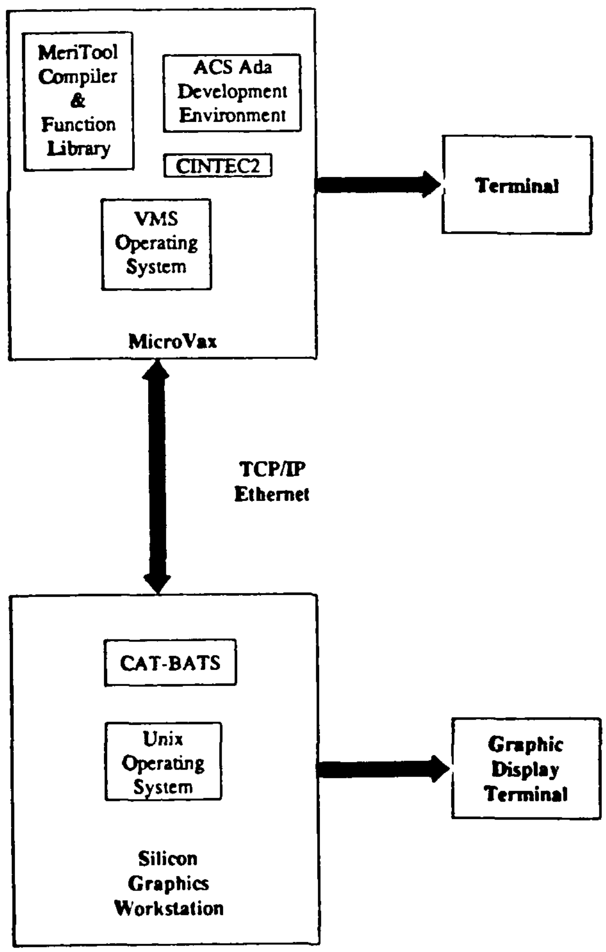


Figure 17 - Simulation Configuration

3.1.2 Simulation Hardware

The E2C2 hardware consisted of the Integrated Test Bed's Silicon Graphics 310GTXB, a MicroVax III computer system, and appropriate ethernet hardware on each machine.

3.1.3 Simulation Software

E2C2 software used in simulation of the CINTEC2 operating environment consisted of simulation models providing air vehicle, avionics, graphics, threat, and support functions. These functions were provided by the CAT-BATS modeling software.

The CAT-BATS simulation models were run on a Silicon Graphics 310GTXB Workstation and provided a graphic depiction of the simulation through a "god's eye" view and windows showing various sensor, mission, and weapon states. Additionally, CAT-BATS provided data recording and analysis capability for the CINTEC2 through the use of graphical plotting tools and simulation playback. CINTEC2 was run on a MicroVax III computer system and provided textual outputs consisting of informational status messages on a connected VT220 terminal. Other outputs from CINTEC2 took the form of sensor and aircraft system commands directed to the CAT-BATS models via TCP Ethernet.

3.1.3.1 Simulation Control

Simulation control was performed through the Silicon Graphics Workstation CAT-BATS interface software. Control consisted of capabilities to change and control graphic displays, initiate a simulation run, and terminate a simulation run. CINTEC2 control was exercised from a VT220 terminal connected to the MicroVax III running CINTEC2. CINTEC2 control consisted of starting and/or stopping the combat controller.

3.1.3.2 Flight Model Functions

The flight model functions simulated ownship aircraft controls, the aerodynamic properties, and the aircraft engine propulsion.

3.1.3.3 Avionics Functions

The avionics functions consisted of the simulation of onboard sensor systems, weapon systems functions (tracking, weapon flyout, and stores), and electronic countermeasures. Table 1 depicts the various onboard systems that were provided for CINTEC2 control.

TABLE 1 - Aircraft Sensor and System Models

Terrain Following Radar	Multi-Mode Radar	Missile Warning Receiver
Radar Warning Receiver	Launch Warning Detector	Optical Sensor
Forward Looking Infrared	Laser Ranger / Designator	Radar Altimeter
Maverick Missile	Paveway II Missile	Barrage Jammer

3.1.3.4 Support Functions

The support functions consisted of simulation models contained within the CAT-BATS simulation system that were not a part of the aircraft modeling. The support functions included aerodynamic models of missiles in flight, ground threats, electronic warfare effects, relative geometry calculations between simulation objects, atmospheric modeling effects, and simulation data collection.

3.2 Test Program

The CINTEC2 Test Program is a program designed to test the effectiveness of the combat controller under a specific set of mission conditions. In particular, a low altitude covert penetration strike mission was chosen for the testing scenario. Ultimately, what is being tested is the value of the decision strategy in providing improved mission effectiveness through integrated control of offensive and defensive aircraft sensors and systems. This is determined through measures of effectiveness (MOE's) that analyze the final results over the course of various scenario executions.

The decision strategy for combat control, as it turns out, is highly dependant upon a wide variety of mission, systems, and environmental factors and hence requires customized decision control strategies in a prototype environment such as that imposed upon CINTEC2. The decision strategy utilized by CINTEC2 is a prototype AI expert system rule set that exploits the known

aspects of the given mission test scenario. The testing scenario was planned so that CINTEC2 will have different problems and conflicts to resolve in order to achieve a successful mission. If other scenarios are to be generated for use with CINTEC2, customization for the particulars of that scenario will need to be incorporated into the decision strategy.

3.2.1 Test Configuration

Since the terrain type was one of the factors effecting mission performance, the scenario was designed to fly over three different terrain areas corresponding to the three terrain types (flat, rolling, rugged). Next, a starting point for the mission and a target was selected. A route was then selected from the initial point to the target. Over rough terrain, the route will place a heavy burden on the TF radar.

Once the route has been established, the threats were placed along the route. The type and density of the threats was varied in order to exercise the rule in different ways. Threat types were varied along the route so that conclusions may be made about CINTEC2 against different threat types in the future. In this prototype version of CINTEC2 threat identification was not performed.

3.2.2 Measures of Effectiveness

Measures of effectiveness were used in the analysis of the simulation runs. Several levels of measures were employed. The lowest level quantified the effectiveness of the controller itself. The next level quantified the effect of the controller on the platform that it is deployed upon. The highest level quantified the effect of the controller on the success of the mission. The following lists the various measures that were determined to be most significant relative to CINTEC2 operation. Not all measures were used in the analysis, however,

3.2.2.1 CINTEC2 Evaluation Measures

Total Number of Operations - The total number of sensor tasks and/or CINTEC2 reactions that occurred. This measure indicates the potential workload of CINTEC2. There may be some runs where CINTEC2 is needed more than others due to different threat densities and types.

Number of Operations Handled Correctly - This is a subjective measure that will need to be calculated by the analyst after the completion of each run. Even though CINTEC2 resolves all problems in some manner, this measure will show how many of the problems were handled correctly. The analyst will decide whether or not the correct decision was made based upon the SA information at the time of the conflict. A timeline of the conflicts may be used by the analyst to do this. This measure will not be used to verify the heuristics already implemented, but rather to determine whether or not new rules need to be implemented.

3.2.2.2 Platform Measures of Effectiveness

Blanking Time - The amount of time the platform's warning receivers are inoperable (i.e., "blanked") due to its own interference. If the warning receiver is blanked for a significant amount of time, the platform may not have enough time to respond to threat activities.

RF Energy Output - The total amount of RF energy (power x time) produced by the platform. Ideally, the covert platform will not have to emit at all. If it does, this measure will quantify the number and intensity of the emissions.

Threat Exposure Time - The amount of time the platform is detected/tracked by a hostile SAM. Although it may be possible for the aircraft to avoid all detection, this may not be realistic. This measure will quantify how well the aircraft remained covert.

Number of Threat Missiles Launched - The number of threat missiles launched against this platform. This is a measure of how well the platform concealed itself from the enemy SAMs. The number of threat missiles that impact will be reflected in the operational measures (probability of survival).

3.2.2.3 Operational Measures of Effectiveness

Probability of Survival [or P(s)] - The cumulative probability of survival of the aircraft at the end of the run. The aircraft's probability of survival will be reduced accordingly for each lethal SAM engagement. For instance, if an aircraft with a P(s) of 1.0 is hit by a missile with a P(k) (Probability of Kill) of 0.2, the aircraft's new P(s) will be 0.8. If the same aircraft is hit by another missile with a P(k) of 0.3, its new P(s) will be 0.56 (0.8-(0.8*0.3)=0.80-0.24=0.56). If the aircraft's probability of survival falls below a preset threshold, the simulation may be stopped.

Weapon Release and Target Closest Approach - The actual distance to the planned weapon release point and/or target that the penetrating aircraft achieved.

Bombs Dropped on Target - The number of bombs dropped and the distance from the target that the bombs landed as released by the penetrating aircraft.

No single measure of effectiveness will be able to determine how effective CINTEC2 is. Only by analyzing all of the measures of effectiveness can the controller be evaluated. The lower level measures of effectiveness should drive the results of the higher level measures of effectiveness. In other words, by resolving sensor conflicts, the CINTEC2 decisions should increase mission success.

This is not to say that CINTEC2 will always ensure a successful mission. For example, in a covert penetration run with CINTEC2, the aircraft may elect to return to base without dropping any bombs on the target. In the same run without CINTEC2, the aircraft may continue on and thus get shot down by a SAM. Both missions were unsuccessful as far as destroying the target, but at least the aircraft was able to survive the mission by utilizing CINTEC2.

3.2.3 Scenario and Initial Conditions

The test scenario, as mentioned previously, was a low altitude covert penetration strike mission. The mission provided the controller with a variety of problems that would need to be handled in one way or another due to the nature of the events and their timing during the mission. Five different and distinct mission phases were supported by the CINTEC2 prototype. These phases were cruise, ingress, target acquisition, target attack, and egress. The decision strategy was planned and the rule set was constructed so as to utilize the given mission phase in the determination of the various sensor and system actions that could be taken.

Terrain effects on the controller were also tested using identical waypoint and threat laydowns over various flat, rolling and rugged terrain segments. Weather obscuration was considered as an input to the scenario also, but after examination of this particular parameter, it was determined that its contribution to the testing of CINTEC2 effectiveness was not significant and that the other factors such as terrain would serve to test the same type of mission environmental effects.

The scenario was customized to provide several problems for CINTEC2 to resolve. These problems, relating to the operational conflicts of sensors due to conflicting emissions between systems and/or the ability to use different sensors to accomplish similar tasks should a particular system be occupied were the main objectives of the scenario testing.

During the scenario, threat engagements consisted of both single and multiple threat encounters. These encounters were forced upon CINTEC2 in order to test its ability to take actions should its overall strategy of remaining covert be disrupted. Even though the overall decision strategy calls for a "covertness priority", it is not reasonable to expect that in a real world theater of engagement that this will be able to be accomplished. For this reason the scenario acts to test the controller's ability to deal with threat systems in such a way that it can regain covertness without disrupting the overall mission.

The scenario used for testing the CINTEC2 software is depicted in Figure 18. Mission routing, waypoints, and threat laydown were all planned in such a way as to exercise the CINTEC2 decision strategy.

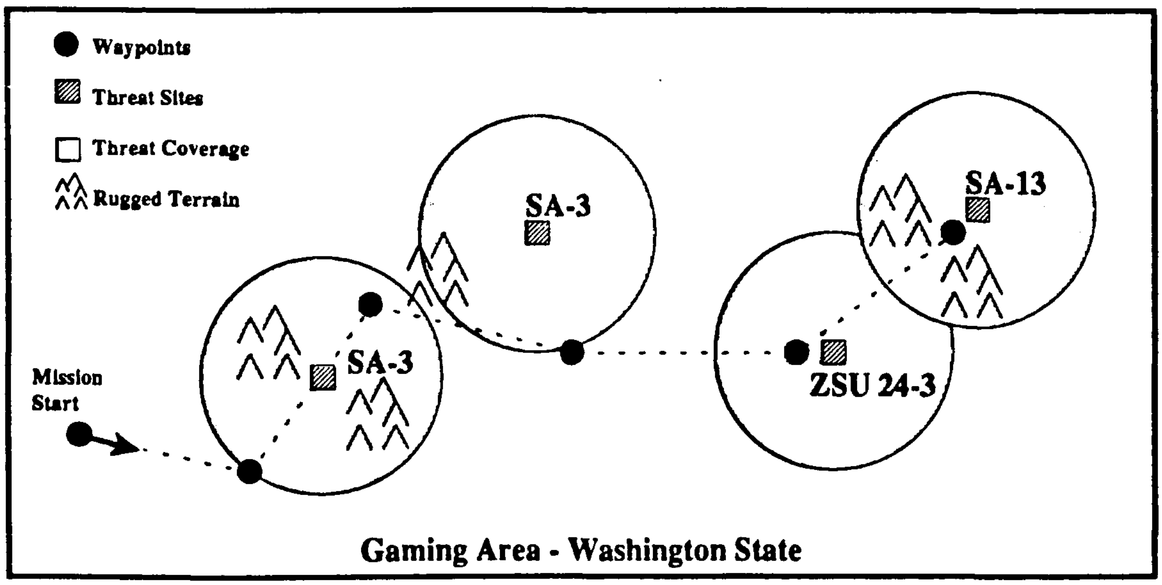


Figure 18 - Test Scenario Diagram

3.2.4 Test Conduct

The scenario testing was conducted at the Wright Laboratories Integrated Test Bed Facility. Three sets of separate executions were performed in order to assess CINTEC2's mission contributions. These executions were performed on identical low level covert penetration strike scenarios. The executions were identical in all respects to mission routing, flight path altitude trajectory, and threat laydown. The differences between the scenario execution runs are listed in table 2.

Table 2 - CINTEC2 Test Runs

Test Set #	CINTEC2 Mode	Offensive Mode	Defensive Mode
1	ON	Controlled TFR	Controlled Jamming
2	OFF	Continuous TFR	Continuous Jamming
3	OFF	Continuous TFR	None

These three basic test sets provided for a CINTEC2 test run and two baseline test runs (test sets 2 & 3) with which to compare CINTEC2 operation. Since the interest in CINTEC2 is to provide a means of remaining more covert while performing sensor and system deconfliction, the most obtrusive components (i.e. jammer, and TFR) were chosen as the system parameters to be varied and tested against. Other avionics systems such as radar altimeters, chaff, and flares can be estimated from the test runs. The reason for this is that the operation of and the results obtained from these other systems will be similar to the results observed with the jammer and TFR due to similarities between their respective operations (when to use, how to use, etc...). Other systems controlled by CINTEC2 provided additional sensor systems with which CINTEC2 had to work with and around in order to provide other necessary sensor operations. The complete list of sensors and systems supported is listed in Figure 4 in Section 2.0.

Test set #2, where continuous jamming was utilized to reduce the ability of the threat system to maintain track on the aircraft and reduce the exposure is indicated by the statistics in Table 3 of the next section. This test set served as the measure for the best attainable performance based upon 100% effective jamming during threat encounters. The only time threats were able to obtain a lock was when the host aircraft was close enough for the ground based radar to "burn through" the jamming effects. In this particular instance, covertness was completely destroyed by the operation of the jammer due to the continuous mode of operation.

During this run it was also necessary to operate the TFR continuously to maintain ground clearance. We assumed that the jammer and TFR did not conflict under this set of circumstances either through interleaving or different frequency usage.

In test set #3, where only the TFR was on continuously, covertness was also degraded due to the continuous operation of the TFR system although not as significantly as with the jammer. Correspondingly, there were a great deal more lethal threat encounters since no countermeasure actions were taken. This test run provided a measure of what the worst performance of the CINTEC2 system would be if no sensor system actions were taken.

The results of the testing were composed of information relating the differing types of threat exposure, probability of survival, and cumulative radiated power from specific aircraft emitters. Specific areas of interest during the tests were how much time the aircraft was exposed to all types of threat radars, the number of acquisitions, tracks, and launches a threat was able to perform on the CINTEC2 aircraft, RF sensor blanking time, cumulative probability of survival, and the total number of CINTEC2 operations over a given time span. These statistical measures are presented and discussed in the tables and figures presented in the next section.

Note that aircraft maneuvering strategy was a CINTEC2 rule set consideration for the puposes of avoiding or escaping threat systems. It was decided, however, that at this stage of the E2C2 program, it was more important to concentrate on the processes involved in sensor deconfliction by eliminating the maneuvering variable from the overall CINTEC2 equation. This would force more sensor and system actions to be taken to counter the threat effects.

3.4 Test Results

This section summarizes the results of the CINTEC2 scenario testing and includes observations as to the overall assessment process and its value to estimating the functional performance of CINTEC2. The collected CINTEC2 test data about which this section is written is included in Appendix A. The measures and quantifications derived from the data is presented in this section. This includes graphs and charts of performances as well as tables of different measures that were examined.

3.4.1 Threat Exposure Analysis

The threat exposure data indicates an improvement in the ability for the CINTEC2 host aircraft to remain more covert through the approach taken to mediate the sensor systems. Although the table indicates that improvement is not dramatic relative to the baselines, the degree of the improvement is a product of the decision strategy employed in the expert system as well as the types of systems available for control.

In examining the threat exposure data, we must analyze the results from two different aspects. The first is to assess the controllers ability to remain covert or completely unexposed to any threat system. The second is to assess its ability of trying to regain a covert state once exposed. In testing these two abilities, it should be noted that in all of the test engagements the aircraft was forced to fly a trajectory through threat environments where threat exposure was unavoidable. Table 3 below summarizes the threat exposure and encounter data for the different simulation test sets.

Table 3 - Threat Exposure Measures

Test Set	Exposure Time (sec)	Exposure Time %	# of Acquires	# of Tracks	# of Launches	# of threats Detecting
1 CINTEC2	193	35%	13	18	6	4
2 Jamming	78	14%	3	8	4	3
3 TFR	194	35%	10	19	7	4

In attempting to remain covert it is crucial to observe that the overall time the CINTEC2 host aircraft was exposed, tracked and/or launched on by a threat system was approximately the same for the runs with and without CINTEC2 in the absence of continuous jamming (Test Sets 1

& 3). Since the definition of remaining covert in our particular instance is to employ only non-emitting sensors and systems, the decision strategy was programmed with this in mind and was successful in this operation. During time frames when the aircraft was not detecting any threat systems only passive systems were permitted to operate as shown in the sensor timeline diagram (figure 19) unless mission phase dictated otherwise. Included in the operation of passive sensor systems were the radar warning receiver, missile warning receiver, launch warning detector, FLIR and optical systems that did not require energy emissions. This passive monitoring state was employed through the decision strategy until CINTEC2 detected a track radar.

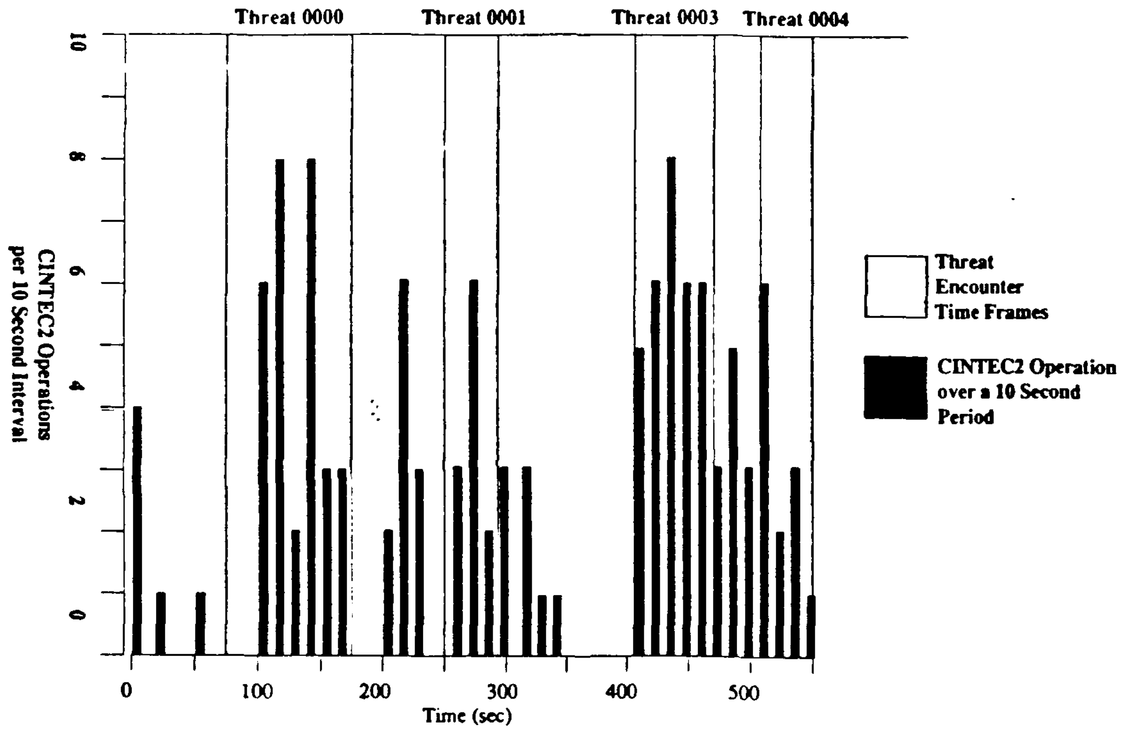


Figure 19 - CINTEC2 Sensor Operation Timeline

Once a track radar was detected by CINTEC2, a new decision mode was entered. This mode was based upon low probability of intercept (LPI) operations that could help the host aircraft regain a covert state. The allowable LPI operations included operation of the TFR and radar altimeter to achieve a lower trajectory profile and hopefully, the line of sight of the tracking threat system(s). The passive sensors were also permitted to operate during this time providing that their operation did not conflict with the operation of the TFR and radar altimeter. Since conflicts did exist between the RF emissions of the TFR/radar altimeter and the passive RF warning sensors, the operation of the systems were interleaved so that CINTEC2 could gain a brief glimpse of the external situation while maintaining its TFR operating mode. If the brief

glimpse of the situation showed that the threat track radar was no longer present, CINTEC2 would attempt to regain its passive operating status. If the track radar was still present, a fully active emission state would be entered where the controller would permit any necessary actions including jamming countermeasures.

The active emission state operated similar to the LPI state of the controller. That is, it would interleave the jammer (active emitter), TFR (LPI emitter), and operation of the passive RF sensors in order to maintain the low altitude TFR profile, effectively jam the threat system, and take quick glimpses of the external situation. Once again if the track radar was no longer seen as operating, the controller would attempt to step back into an LPI state, and eventually back to a passive state when and if this became possible. During the operation of the particular sensor activities required in the various emission states, other non-conflicting sensors such as FLIR, optical, and laser designator were mediated on an as needed basis by the controller. This overall covert decision philosophy has indicated that the vast majority of sensor mediation operations occurred during the threat encounter portions of the missions (this can be observed in the Figure 19 diagram). The operations of CINTEC2 in the regions not concerning threat encounters represents sensor operations that are necessary for passive navigation updates, LPI operations during different mission phases, and manipulation of the passive RF warning sensors in conjunction with LPI operations.

In attempting to regain a covert state, CINTEC2 shows improvement over the test set 3 baseline by reducing the number of tracks and launches that threats were able to perform. Although the observed improvements are not dramatic, more improvement could be achieved through further refinements of the decision strategy to employ additional threat defeating strategies based on actual and perceived threat system knowledge. Additional threat defeating techniques such as maneuvers and countermeasures would also increase the performance of CINTEC2 on the threat environment. Even so, the CINTEC2 manipulation of the defensive RF jammer in conjunction with the TFR and passive RF warning sensors demonstrates that a reduction in threat lethality can be achieved without excessive emissions. It is interesting to note that the number of threat acquisitions actually increased due to the ability CINTEC2 had to confuse the threat radar through use of the jammer. This actually indicates that the threat was having to search for the target more often and points to CINTEC2's attempts to regain a covert state. These statistics were summarized in Table 3.

3.4.2 Power Emission Analysis

The overall energy emissions observed in the three test set runs indicate a substantial reduction in the power emitted by the CINTEC2 mediated sensor systems compared with the baseline test sets without CINTEC2. This is a direct measure of improved covertness. In addition, even though blanking of the passive RF sensor systems was unavoidable, CINTEC2 was able to reduce the effects of the blanking by interleaving the passive and active sensor commands. This resulted in the passive RF sensors being given brief "glimpses" of the external situation and passing the information into CINTEC2 for processing. Table 4 below indicates the various blanking times and power emissions for the various test sets.

Table 4 Radiated Power and Blanking Measures

Test Set	Radiated Jammer Power (W/h)	Jammer Operating Percentage (%)	Radiated TFR Power (W/h)	TFR Operating Percentage (%)	RF Sensor Blanking (s)	RF Sensor Blanking (%)
1	2.2	14%	4.3	31%	316	57%
2	15.3	100%	15.3	100%	551	100%
3	0.0	0%	15.3	100%	551	100%

The total emitted power for test set 2 was the greatest at 30.6 W/h. This was a combination on the TFR/radar altimeter and the jammer systems which were on continuously during this run. Only running the TFR/radar altimeter resulted in 15.3 W/h being emitted and finally, with CINTEC2 usage, only 6 5 W/h were emitted. The reduction in power emissions can be directly correlated with the CINTEC2 "desire" to remain covert, and, if no threat encounters had taken place, no emissions would have been radiated. Radiated power for the jammer and TFR were set at 100 W to simplify the power output comparisons.

3.4.3 CINTEC2 Operational Analysis

The CINTEC2 software, on a computational scale performed very few operations. In fact, a total of 130 operations were conducted by the software in a 9-minute time frame and most of these operations occurred in bursts brought on by encounters with threat systems. Computationally speaking, just about any computer processor could perform this task. Over the 9-minute mission segment CINTEC2 averaged one operation every 4 seconds or about .25

operations per second. During the threat encounters operations increased to 1 operation every 2.5 seconds or about .40 operations per second. In addition to the actions taken by CINTEC2, all systems were continuously monitored which the pilot would not be able to do. These statistics are summarized in Table 5.

Although the numbers of operations performed by CINTEC2 were not large from a processing standpoint, if performed by a pilot they would be overwhelming and extremely tedious to perform. A demonstration of the CINTEC2 software showing the switching of the various avionics systems graphically demonstrated this fact. The vast majority of these operations were performed due to mediations between the TFR/radar altimeter, the jammer, and the passive RF receivers in competition for time to do their respective tasks. Judging from the level of the improvement in threat exposure assessed in the previous section, it would be beneficial to add other systems that could further improve the system performance. These types of systems fall into the categories of those needed for applying additional countermeasures, preemptive weapon delivery, and/or radar functions, etc. These systems in turn will add additional conflicts and need mediation in order to operate effectively and this will further increase the numbers and types of CINTEC2 operations that must be performed.

One further point is that the numbers of operations for the baseline test set 2 and 3 can only be determined through a man-in-the-loop type simulation where the pilot performs given system commands in whatever manner necessary to accomplish his goals. This would provide a measure of pilot sensor control performance and effectiveness relative to that employed by CINTEC2. The pilot mission runs to accomplish this would need to be held identical to CINTEC2 with the pilot having the same resources available that CINTEC2 employed. Even so, it is important to remember that there are a myriad of other factors that can be brought into the CINTEC2 control scheme as well as factors that will demand the pilot's attention.

Table 5 - CINTEC2 Operational Measures

Test Set	Total # of CINTEC2 Operations	Overall CINTEC2 Operations / sec	Threat Encounter Operations / sec	Missile Survival Probability based solely on Jamming
1	130	.25	.40	30%
2	x	x	x	48%
3	x	x	x	23%

The last measure of effectiveness that needs to be examined is a combination measure that relates all of the different event occurrences during the mission into one statistic. This is the cumulative probability of survival which is also summarized in Table 5. Based upon the three test sets, the probability of survival increased from the basic simulation baseline execution made without the CINTEC2 controller and no jammer (test set 3), but decreased relative to operating the jammer continuously (test set 2). The primary reason for this relationship between the test set survival probabilities is directly related to the number of missile launches that occurred during the respective test scenario executions. Because CINTEC2 was able to effectively defeat or prevent one missile launch as compared to the test set 3 baseline, the probability of survival increased relative to test set 3. Because the continuous operation of the jammer (at the expense of any covertness) prevented three launches as compared with test set 3, it's probability of survival increased even more. The exclusive use of a jammer under these circumstances as the only countermeasure drives the CINTEC2 probability of survival lower than could be realized by the addition of other systems and tactics. The addition of chaff, flares, maneuvers, and pre-emptive strikes will dramatically increase the CINTEC2 effectiveness and the probability of survival. Even so, CINTEC2 was still able to show improvement through the employment of one countermeasure system.

It is important to note that these test sets did not take into account any capabilities of the ground based threat systems to employ passive receivers in attempting to track and launch on the CINTEC2 host aircraft. This factor alone would decrease the success observed in the test set where continuous jamming is employed. It represents an area that needs to be investigated at some point in order to give a better baseline test with which to compare CINTEC2 test results. As mentioned previously, however, covertness is non-existent in this scenario.

The rest of this subsection is composed of charts of the data collected during the different test set runs along with a brief description of the data of interest on the charts. They are provided to show the effects that CINTEC2 had on the simulation during its test run compared to the two baseline test sets. Figures 20, 21, and 22 represent data collected during the three different test set executions respectively. The graphs depict threat exposure time, terrain masking and terrain following profiles. Statistics are also summarized at the bottom of the threat exposure and terrain masking profile graphs which represent, acquisition, track and launch statistics as well as cumulative exposure and percentage exposures. The statistics were summarized previously in Table 3. In comparing the three different runs it is important to notice that the flight trajectories and terrain maskings were identical and that threat exposure was different for given test sets, however only slightly between test sets 1 and 3.

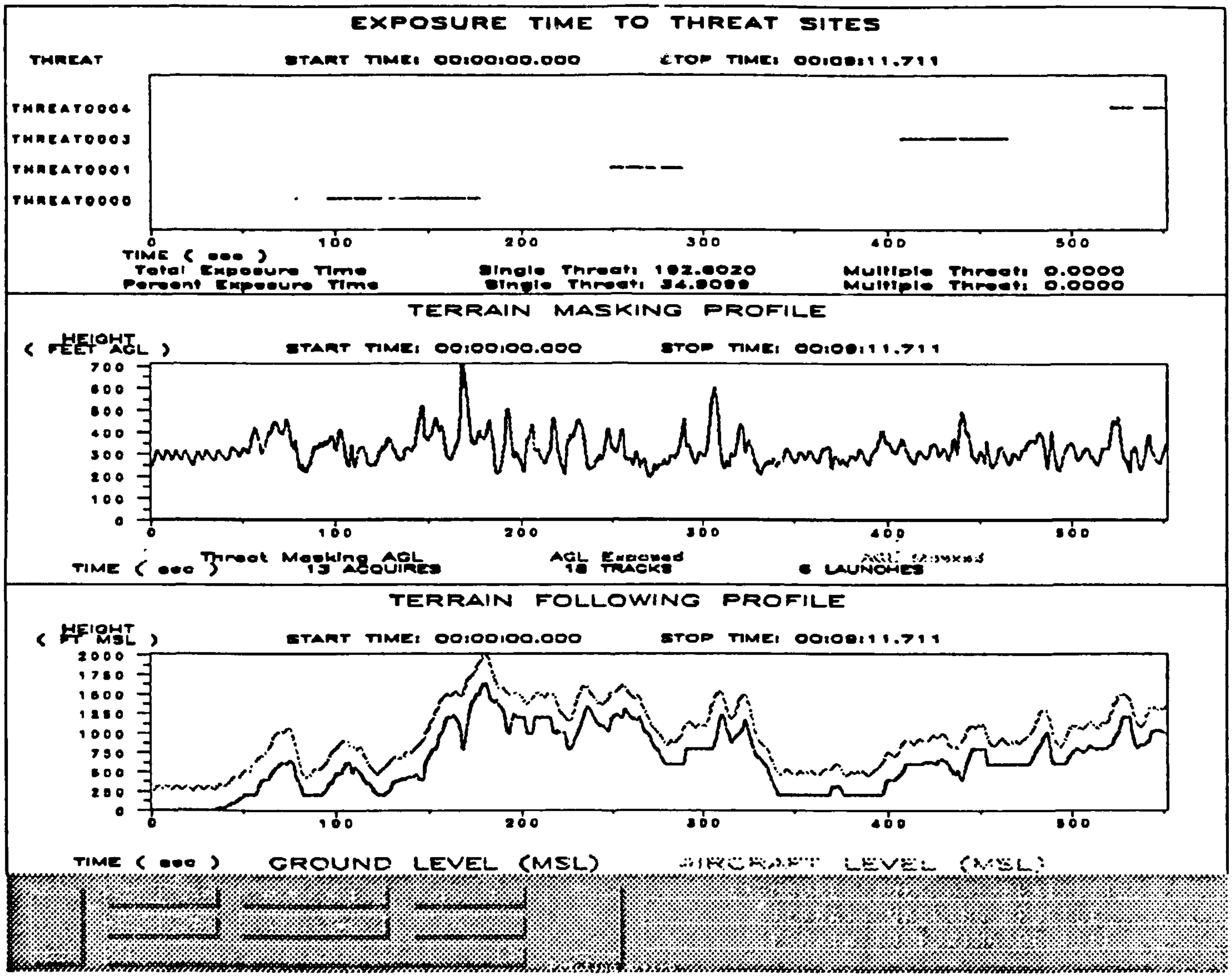


Figure 20 - Test Set #1 - CINTEC2
Threat Exposure and Terrain Profiles

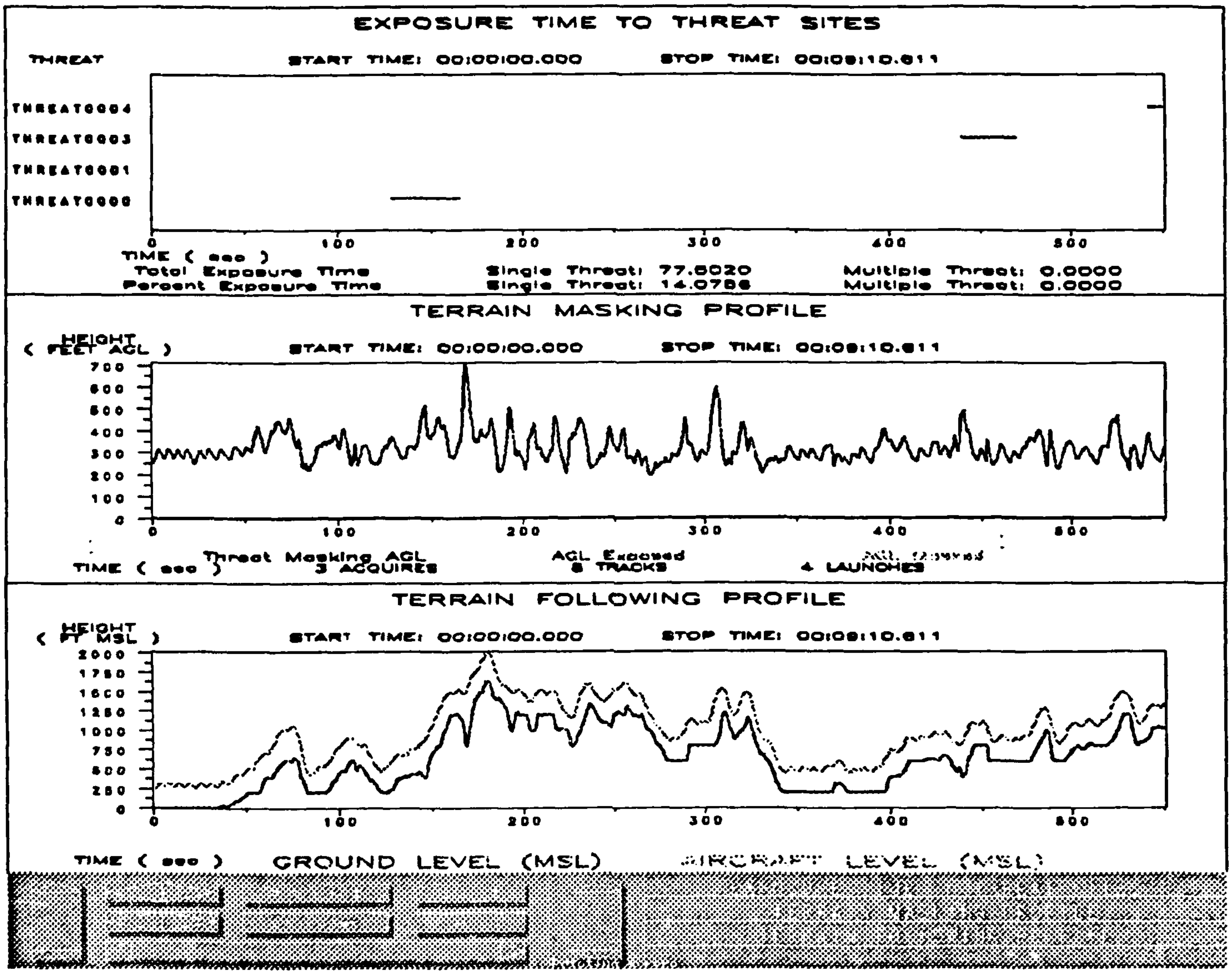


Figure 21 - Test Set #2 - Continuous Jamming, Continuous TFR
Threat Exposure and Terrain Profiles

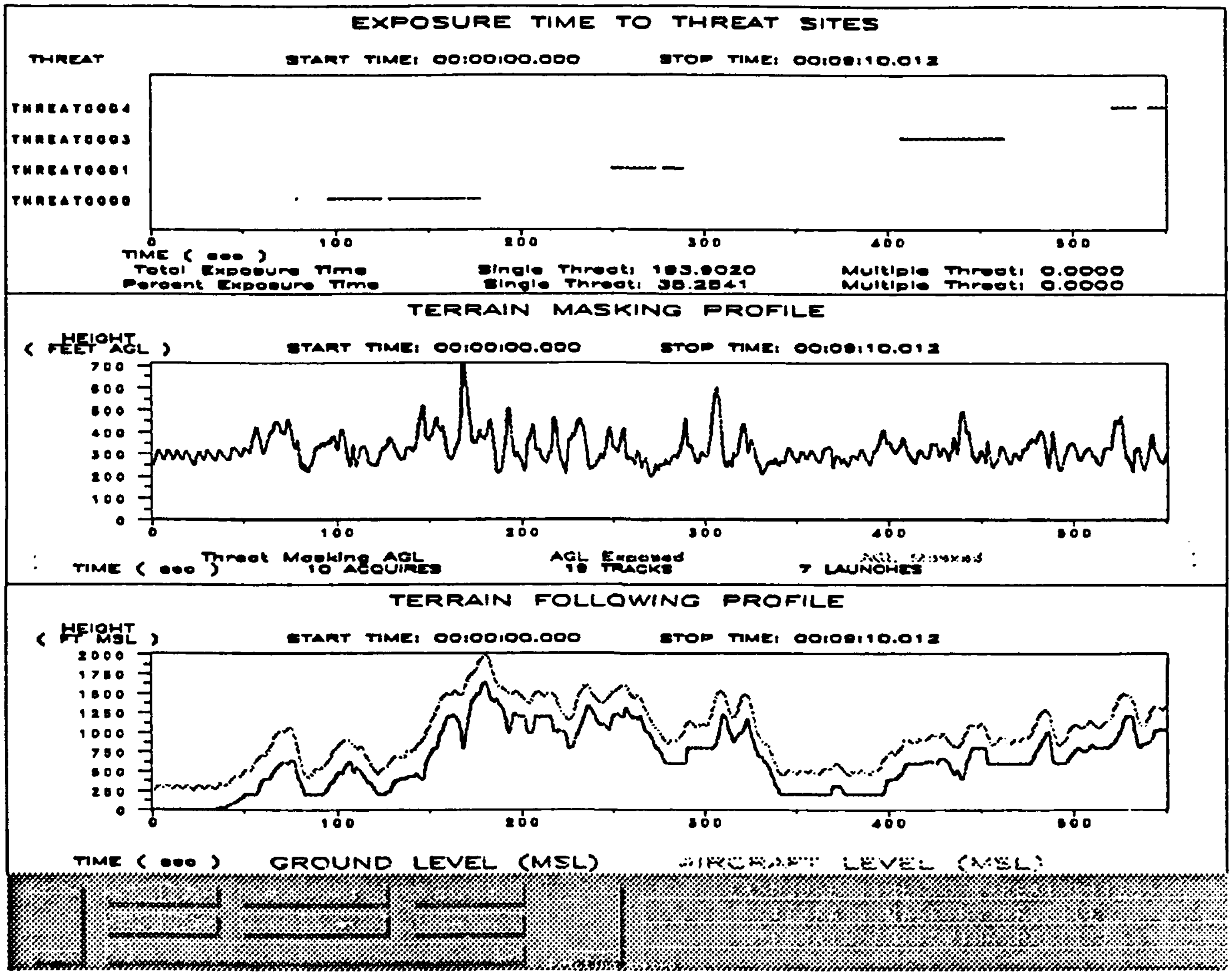


Figure 22 - Test Set #3 - Continuous TFR, No Jamming
Threat Exposure and Terrain Profiles

Figures 23, 24, and 25 represent the detected signal strengths and signal to interference ratio observed during the first CINTEC2 threat encounter (simulation threat 0000) for the three test runs. Important to note in these charts is the oscillation of the signal to interference ratio that threat 0000 perceived during the CINTEC2 test run. This was due to the continuous system mediation that CINTEC2 performed which resulted in the turning on and off of the jammer. The two baseline test runs show a peak in the SNR at the closest point of approach that the aircraft made to the threat, although the peaks are at about 30 dB when the jammer was off (test set 3) and 115 dB with the jammer on (test set 2). The detected signal strength graphs show where the threat was able to achieve track by displaying a small block on the signal strength line. Blocks that appear below the line indicate that a track could not be obtained due to jammer effects on the ground based threat system.

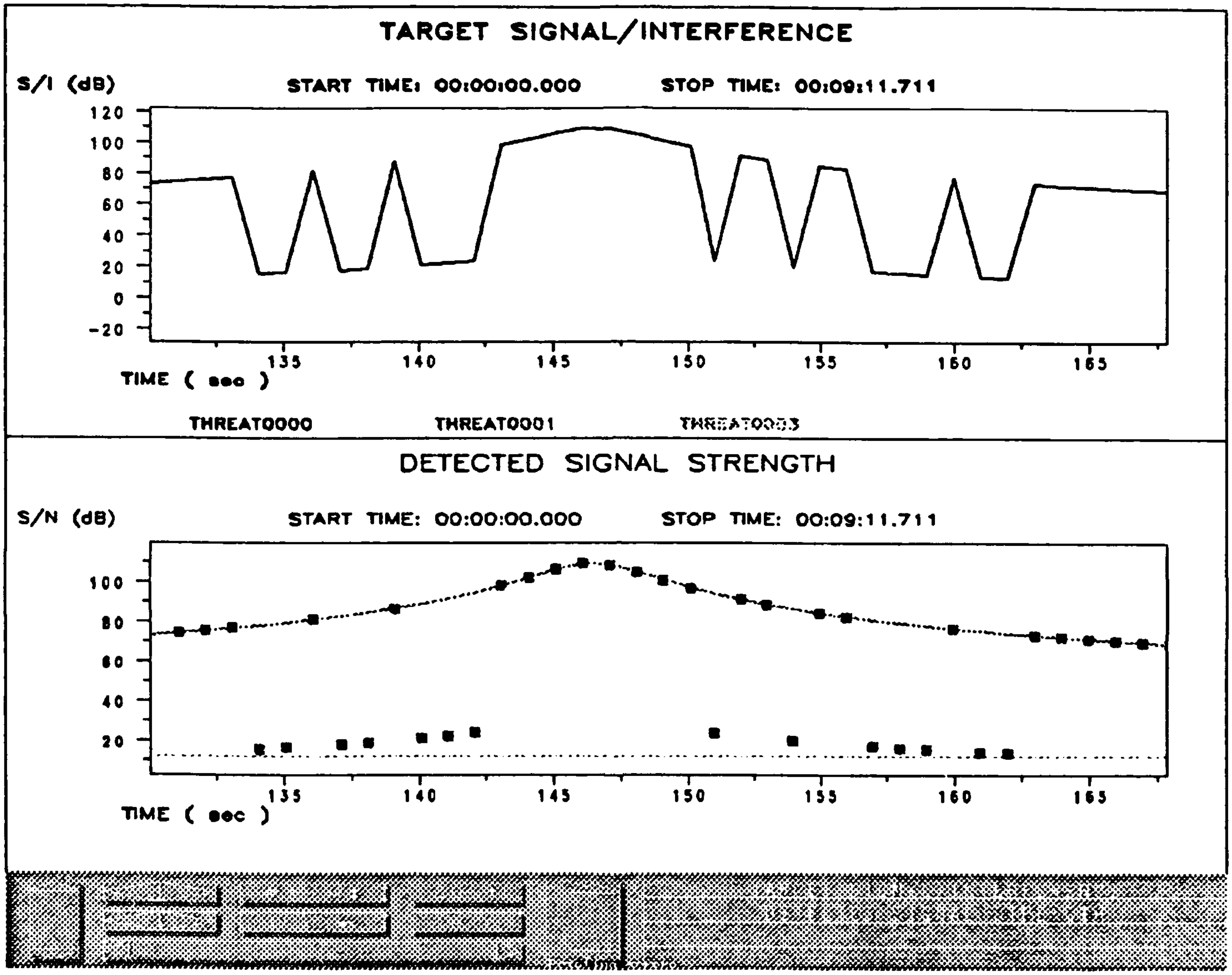


Figure 23 - Test Set #1 - CINTEC2
Signal Interference and Detected Signal Strength

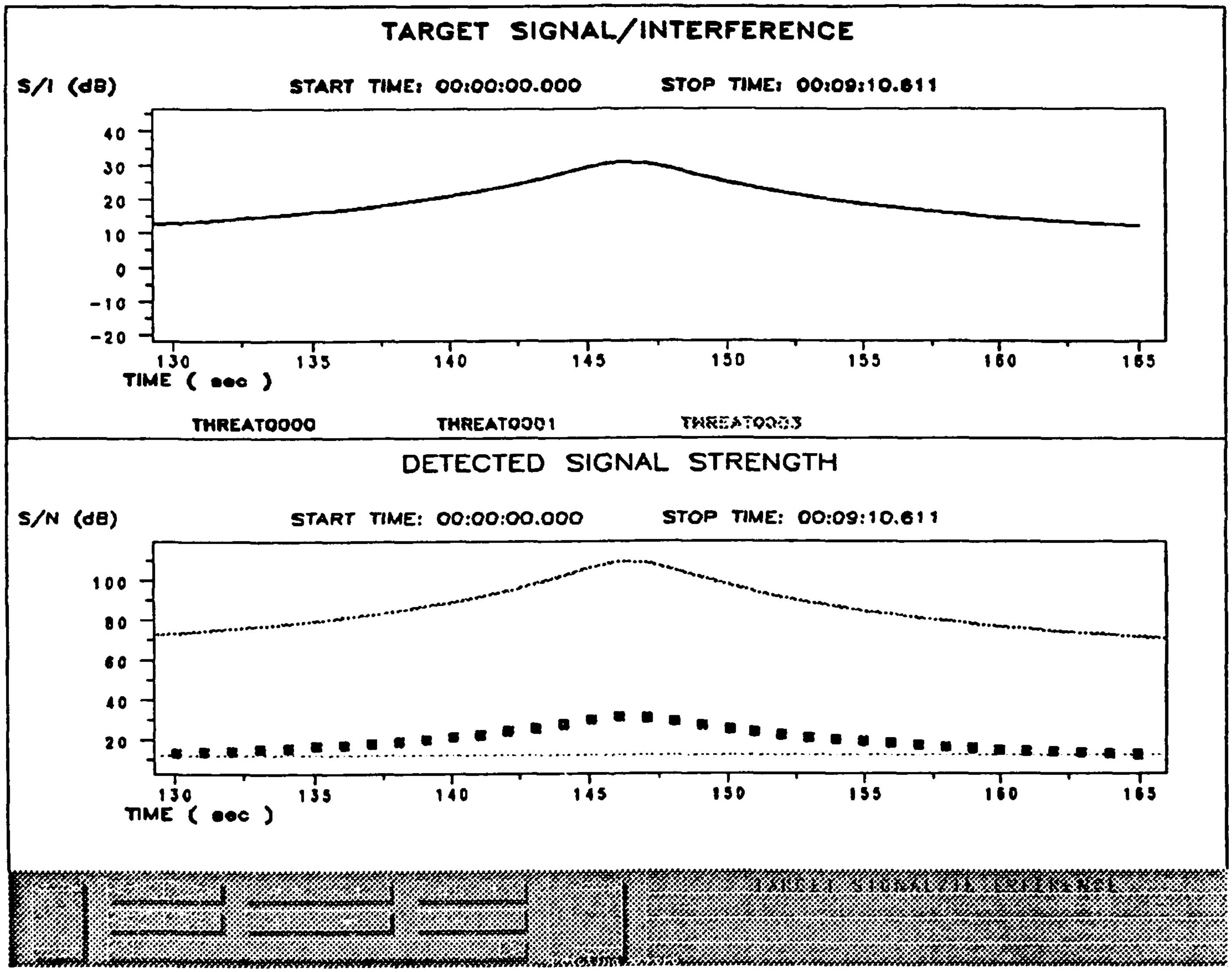


Figure 24 - Test Set #2 - Continuous Jamming, Continuous TFR
Signal Interference and Detected Signal Strength

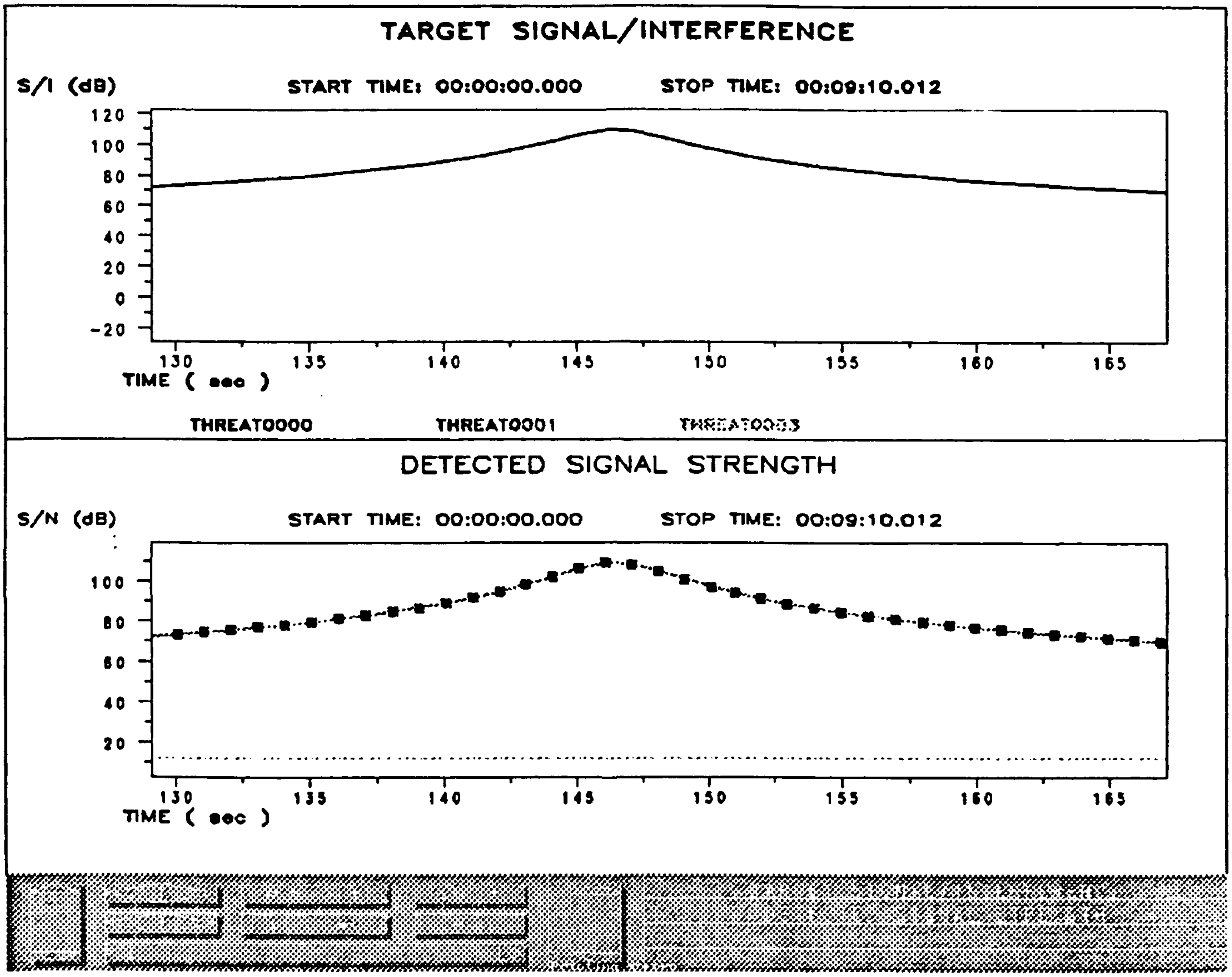


Figure 25 - Test Set #3 - Continuous TFR, No Jamming
Signal Interference and Detected Signal Strength

3.4.4 Assessment of CINTEC2 Results

It is very difficult to assess the particular advantages and benefits of a system such as CINTEC2 in a purely objective context without taking into account the myriad of different factors that can effect the test results. Since the prototype examines only a subset of the possible inputs to such a system, a "true" measure of its performance is impossible. What can be gleaned from the testing, however, is the benefit of the controller in certain areas relating to known operational problems of sensors such as blanking or the measure of improvement of some desirable quantity such as a decrease in threat exposure. Even in this context, however, test results can be misleading if examined only at face value.

Synopsis of the test results indicated that with CINTEC2 turned on, threat exposure dropped and sensor blanking time was reduced relative to the baselines. In addition, RF energy emissions were reduced and threat missile launches on the ownship were also reduced. Of course, even though these results indicate a performance improvement, it is important to remember that the simulation models will respond directly and immediately to the changes that CINTEC2 makes in controlling the systems and that these models are limited in nature. However, it is also important to note that the limited set of circumstances presented to and handled by CINTEC2 indicate that expert system control and mediation of systems can represent a means of improving the survivability of aircraft if the decision strategy is made robust.

The results of the scenario testing are subject to the performance of the models used to define the aircraft, sensor models, threat systems, and other avionics. For this reason it is important that the models be as true to life as possible in order to ascertain accurate estimates of the performance. The CAT-BATS models are very good for testing in some respects, but not so good in others. From a prototyping point of view, the CAT-BATS simulation used to develop and test the decision strategy and accompanying Ada software components provided an excellent vehicle with which to assemble and exercise ideas. Changes could be brought about quickly and tested in a matter of minutes. The drawback came during the testing of the controller because the results were effected by the fidelity of the models to some degree. The degree of effect on the results is somewhat subjective in nature, but it appears to improve the overall performance of the controller.

In addition to this, measurements of certain performance aspects as well as statistics relating the overall "success" or "failure" of the system to handle the different sets of circumstances tell only a very narrow part of the story. There are two reasons for this effect on

the objective results. The first is the nature of the prototype software and the second is that scenario and decision strategy relationships are very closely intertwined.

The nature of the prototype software is to explore the concepts of the combat controller principle by deriving advantage from integrated offensive and defensive sensor and systems control. Since we are dealing in the prototype domain, we are limited in the amounts and types of information that can be brought to bear upon the problem. If we examine the problem in terms of this and realize that the ideal situation would be to have all information possible in our decision strategy, we realize that anything less than this may lead our decision maker into an incorrect decision. However, we have found that certain portions of information are far more critical than others. Determination of the utility of the information that is brought to bear on the problem can only be accomplished through the careful examination of the rules that can be invoked to solve given problems.

The second reason that also has an impact on the objective nature of the results is the close relationship between the controller decision strategy and the scenario events. This is because for a given set of circumstances that may be generated by constructing a test scenario in a given fashion, the knowledge of those circumstances can be used to exercise correct responses of the controller based upon the apriori knowledge. At some point, therefore, it becomes necessary to test the controller with the injection of unknown factors that can influence the decision strategy in order to clearly determine the effects of such realistic occurrences.

4.0 Summary and Conclusions

This section documents the E2C2 program findings, conclusions, and lessons learned as they relate to CINTEC2 and the control of aircraft offensive and defensive sensors and systems. The CINTEC2 software development and scenario testing was geared toward demonstrating and optimizing the use of aircraft functions relating to sensors and systems while taking into account the various mission, sensor, and environmental interdependencies that are involved. The summary and conclusions presented as a part of this section elaborates on the advantages and utility of the use of this information in the performance of integrated control of various offensive and defensive aircraft systems.

4.1 Operational Utility Conclusions

The utility of the CINTEC2 controller lies in its ability to take information from various sources and interpret this information in such a way as to dynamically account for both internal and external events. The artificial intelligence basis for the controller allows for the interpretation of a highly diverse set of data inputs, each which can be used in many advantageous ways. Data characteristics have no effect on CINTEC2 as the information presented to it can be interpreted into rules in any way necessary and thus are not subject to the constraints of a purely analytical approach.

4.2 System Benefits

In the final analysis, CINTEC2 can offer great advantages through the integrated control of sensors and systems of all types. The prototype successfully demonstrates the potential value of such a system. It can provide timely analysis, warnings, and advisories while reducing and or alleviating the potential hazards associated with anticipated or spontaneous sensor blanking and or sensor conflicts. Even in an advisory capacity the decision analysis performed by CINTEC2 can offer valuable and timely information to aid in the selection of sensor, avionics, and/or weapon systems. This information can serve to assist in avoiding threats, reduce the lethality of threat encounters, and reduce the pilot workload.

These test results are only an indicator of how much potential CINTEC2 can actually offer through expert system control. Many more rules, scenarios, information sources, and

mission considerations could still be brought to bear upon the integrated control of systems. Navigation and communication issues as well as other systems could be integrated through the CINTEC2 venue. The expert system approach offers a significant advantage in integrating the various sensor and system components together in a cohesive way that permits the effects of different systems to be accounted for in all aspects of mission operation. The CINTEC2 rules are easy to construct and require only an external data interface to the particular system desired in order to fully integrate the component. Once integrated, the rules for the different control philosophies that can be employed can be easily modified to "fine tune" the particular system operations for maximum effectiveness.

4.3 Lessons Learned

This section documents the lessons learned during the E2C2 program as they pertain to the areas of algorithm and system design, expert system processing, simulation, and most importantly, the integration of offensive and defensive avionics systems.

4.3.1 Algorithm/System Design

(1) Algorithm and system design for CINTEC2 was performed using object oriented design and coding techniques in the Ada programming language. This methodology has pointed out both good and bad aspects of this type of approach. For the most part, however, these techniques offer a significant advantage over the conventional top down structured function oriented approach. The lessons learned in this area pertain to design issues and error handling.

(2) With respect to algorithm and code design for CINTEC2, it became very easy to construct, integrate, test and debug the code from an object oriented design philosophy. The design itself however was somewhat more difficult to perform. This was the case because all of the details of the object oriented algorithms, including the data structures, must be completely and thoroughly defined before any of the coding may be performed. Although this may be standard practice in any design and coding effort, in actuality some details inevitably squeeze through the cracks and are taken care of at coding time. With object oriented programming this can result in disaster. The reason for this is that many different sections of the algorithms will make use of the same data objects which in turn may be built upon even smaller data objects. A mistake in a basic object element can therefore ripple into all portions of the code causing many code changes to interface sections as well as the algorithms themselves.

(3) Due to the nature of CINTEC2 with its Situation Awareness component and EW Planning Component having to share data structures, the needs of both functional components had to be considered in the creation of all of the various data objects. A thorough job here saved many headaches during code integration because as mentioned previously, if one object was not defined correctly, it would effect all other sections of the algorithm that made use of it. Although we had some minor problems relative to this issue, we were fortunate to not have any major oversights. The problems we did encounter, however, pointed out the extreme problems that could potentially be encountered.

(4) The second lesson learned under the category of algorithm and system design concerns error handling. Unlike most languages, the Ada language has an excellent facility for handling all types of errors under all types of circumstances. Errors are not merely detected, they are trapped and presented for error handling. A lot of valuable time can be saved if the error handling capabilities are used with diligence. The facility can prevent uninitialized variables from secretly reeking havoc, protect code from array bounds overflows and other problems far too numerous to mention here. The reason it is presented here as a lesson learned is because even though we put many error handling traps into the code, there could have been many more which would have ended up saving more time in the long run. Because many of the problems we encountered during the integration of CINTEC2 into the CAT-BATS simulation (or any simulation for that matter) were of the specific nature mentioned previously, more attention to error handling will save more hours than it spends in the long run.

4.3.2 Expert System Processing

The expert system processing that takes place in CINTEC2 brought forward some very important aspects of avionics integration both during development and coding of the rule set. These aspects relate to the areas of rule set maintainability, situation awareness input data, the relationship of mathematical algorithms to the expert system, and the overall place an expert system will prove useful in avionics integration.

(1) The AI expert system rule set offers significant advantages in readability and maintainability not only to a programmer, but also to an analyst who may not have any programming experience. The rules can be understood and interpreted easily with only a short introduction as to the overall syntax. This lets mission experts and analysts participate in the actual construction and/or modification of the rules far more easily than could be done using conventional programming techniques.

(2) The ease of modification of the rule set allows for the customization of rules for a particular anticipated set of events or threat laydown based upon current intelligence information. This provides an ability to "fine tune" specific rules to handle specific sets of known or anticipated circumstances. This can be performed without effecting the rules that must always be present to perform certain functions through the use of different rule set partitions. In this way, a basic set of rules can be added to or modified without change to the original rules.

(3) Another advantage of the expert system rule based approach is that it has the ability to deal with extremely different types of information. Whether the information is symbolic (textual), floating point, or integer makes no difference. Rules can be constructed to take advantage of the particular attributes of a certain type of data in the decision making process. This characteristic is extremely advantageous in an avionics environment composed of differing sensors and systems each with their own peculiar data attributes. Messages such as those from external communication sources as well as data from internal aircraft avionics systems (i.e. RWR, etc) can be brought together for integrated decision making on sensor usage, pilot information, and maneuver control.

(4) Another consideration with regard to expert systems in the role of avionics integration concerns its overall function in the avionics environment. While expert systems are capable of making evaluations and decisions towards optimally controlling various systems, they are not optimal in terms of the efficiency requirements for specific low level sensor control. Specific duties such as that of slewing antennas, tracking multiple targets, or performing volumetric searches can better be performed from a mathematical approach for optimal control. This is because these particular tasks have a given set of parameters, all of which are known. The emphasis here being on **known** parameters. Many highly efficient analytical based algorithms suitable for performing high speed solutions need to be used for this type of avionics task. The point at which the expert system advantages come into play is at a level directly above this function.

(5) This point of application is at the level to control the decision making process of what antenna needs to be slewed, which targets are the most important to maintain track of, what algorithm best suits the type of volumetric search which needs to be performed, and what are the limits of specific search areas. In so doing the expert system is capable of examining a vast set of data from multiple sources through which a more robust decision can be made.

(6) Test results indicated that the expert system can consider many more elements of information than is possible by each individual avionics component or by a pilot/EWO combination. Events can be anticipated, sensor conflicts dealt with efficiently, and unnecessary emissions eliminated while maintaining the function of existing avionics components. CINTEC2 has effectively demonstrated the ability to collect and analyze information and improve the performance of modeled avionics systems in this regard.

(7) Although expert systems are a means for generating avionics systems control methodologies, they are best used as a prototyping tool and not as the end product. Even though they can provide an easy method for experimentally trying different control strategies quickly, they are not "trustworthy" enough for the actual implementation on an airframe. The process of inferencing with an expert system is somewhat more prone to error than traditional programming practices at this point in time. Infinite rule executions can be generated very easily even when great care is taken in designing the rule set regardless of how easy the rules are to write. This occurs because a given infinite rule chain may be composed of many rules, some of which may be totally unrelated except through another different rule. These indirect relationships would make it necessary to test every conceivable set of inputs to the expert to be sure that no infinite chains exist. Because of the complexity of the inputs required in an avionics environment, this is virtually impossible.

4.3.3 Simulation

Simulation testing of CINTEC2 revealed some areas that will need to be explored in the future toward applying expert systems to avionics integration. The scope of the E2C2 R&D effort did not permit the testing of the combat controller in a real time man-in-the-loop simulation with independent simulated avionics systems, but only in a canned scenario environment using low-to-medium fidelity avionics models. For this reason, the areas of interest that were left unexplored are concerned with pilot interaction and avionics interfaces.

(1) Testbeds for building and testing avionics integration architectures are difficult to come by in fitting specific program needs. Even when testbeds are available, usage is very dependant on customization that can be performed on the testbed so that requirements for a given program can be met. This can easily represent a significant hurdle to a given program in both time and money especially when the R&D efforts are small. Customization of the testbed detracts from the actual goals of a given program by diverting effort toward supporting functions instead of at the problem under investigation. To some extent, this was a factor with E2C2 but, fortunately

Merit's CAT-BATS simulation was robust enough to alleviate many of the problems typically encountered in this area.

(2) Because a system of the nature of CINTEC2 makes decisions based upon the inputs from external sources, this information could be used indirectly by the pilot or copilot for planning and/or manual control operations. The CINTEC2 information could reveal important events, upcoming tasks, status, or problems in a fused and coherent manner that could be taken into consideration by the pilot if directed to him in the correct way. Although this may sound similar to the work performed on such programs as the Pilot's Associate, the information from CINTEC2 on the whole is of a low level nature not readily consumable by the pilot directly. It could serve as an input and even enhance the operation of a system such as Pilot's Associate, however.

(3) Avionics interfaces, although well-defined and understood from a data passing point of view, do have an important impact on the possible operational considerations of a system like CINTEC2. From the aspect of looking at the outputs of differing avionics systems, whether they are offensive, defensive, navigation, or communication oriented, each has its own piece or pieces of information that can typically be made use of. This is true regardless of the system in question including cooperative systems such as jammer and RWR combinations or reduced power terrain followers that may function in a non-conflicting manner. These types of systems actually help CINTEC2 do its job by providing more capability. Where the bottleneck lies is on the input or control side of the different avionics subsystems.

(4) The bottleneck referred to exists primarily because of the traditional approach to avionics system control. The capability the CINTEC2 can exhibit is directly proportional to the amount of control permitted by a particular avionics system. Autonomous systems that can dispense chaff or flares or perform self-protection jamming, for instance, cannot be easily controlled since they themselves are interpreting the situation and deciding when to take action. Interrupting this action with a system such as CINTEC2 is difficult unless override inputs to the system can be exercised. In the absence of inputs to these types of systems, it is impossible to mediate or deconflict the effects of the actions these autonomous devices will exhibit on other aircraft systems. From this point of view, no control or mediation scheme can be devised that will completely alleviate such a problem. For this reason, avionics need to have robust interfaces so that future integration with other systems can be achieved.

(5) The increase in computer processing power over the last 10 years makes it possible to actually perform expert system sensor control and decision processing in a length of time reasonable enough to offer significant value to actual onboard systems. Our experience with CINTEC2 has indicated that a fairly substantial number of rules can be executed and a decision made within about 1 seconds time. Speaking in terms of system and sensor high level control, this time frame is sufficiently adequate for taking actions on most spontaneous events and may actually be even better than a pilot under workload will be able to achieve.

(6) The final lesson learned with respect to E2C2 was that the information available relative to the performance of sensor systems and their interactions with one another are few and far between. To overcome this problem we made the assumption that like sensors conflicted with like sensors (i.e. RF with RF, etc.). This may or may not be a valid assumption. Consequently, studies need to be made to determine critical sensor interrelationships if E2C2 hopes to exploit various inter-sensor operating characteristics. A data base of sensor conflict information would represent a wealth of possibilities for avionics integration.

4.4 Overall Summary and Prospects for Continuing Efforts

The E2C2 program investigated and analyzed many different areas related to aircraft mission, ground based threat systems, offensive and defensive sensor systems, weapon systems, navigation and communications systems, and their respective current technological impacts and future technological forecasts. This was performed in order to better understand the wide variety of factors that can be utilized in an expert system sensor/system integration scheme. This information was used in the design and development of the various CINTEC2 prototype algorithms for the control and mediation of offensive/defensive avionics systems.

The results of the research and development performed as a part of the E2C2 program and the CINTEC2 development effort have indicated that an CINTEC2 type of system can offer substantial advantages in performing efficient intelligent control of all types of aircraft sensors and subsystems from a systems management perspective. Even though the CINTEC2 prototype was limited in scope to a particular set of events, controls, and inputs, the overall scheme and philosophy of the combat controller is sound and can offer significant advantages when taken to its highest level.

In terms of avionics integration, a high level manager such as CINTEC2 can serve a large role in enabling existing systems to operate within the constraints of specific sensor limitations, system conflicts, and current/future mission strategies. This capability will prove to be invaluable as avionics systems continue to become more robust in their function, demanding more resources, conflicting with more systems, and increasing the workload on the pilot. A system such as CINTEC2 can alleviate many of the problems by providing the methodology needed to successfully integrate systems at a high level.

Additionally, CINTEC2, with its ability to mediate the different avionics systems, is just one step away from being able to act as a channel through which existing or future avionics systems may find it possible to share data. This may require some engineering retrofit of existing avionics systems, but the advantages available through this architecture could be tremendous. Data sharing could reduce the overall avionics burden on the bus architecture, on the pilot, and increase avionics lifetimes by reducing the numbers of system activities that are required to perform different tasks.

The E2C2 program merely scratched the surface in addressing the various issues associated with the integration of avionics systems. Because the program dealt primarily with the avionics and sensor systems associated with traditionally offensive and defensive processes, it left quite a bit of other functionality untouched. For example, the navigation and communication side of the avionics integration equation has not been addressed. This in itself is a significant step in the process of complete systems integration. In addition to this, there are many other types and pieces of information that could be brought to bear upon the integration problem. Information concerning threat system capabilities, additional avionics systems, more robust mission information, and air-to-air engagement information to name a few. Much of this information is available or will be available through current and upcoming avionics technology.

Given that these information sources are or will be available in the near term, it is important to consider their impact on the overall avionics integration picture. Aside from the extreme advantages that the systems will give USAF pilots in successfully performing their missions, they will have the drawback of adding to the complexity of the sensor and system conflict arena. These conflicts, if not managed properly will be catastrophic in terms of avionics system performance and even aircraft survivability.

Existing aircraft have a low level of avionics integration. Each avionics system has its own antenna/aperture and unique signal processor and very little information is shared between the different systems. While dramatic strides have been made in the performance and reliability of individual components, the increasing complexity of the avionics suite as a whole has resulted in complex interrelationships between the avionics systems in terms of operational utility.

A CINTEC2 type of controller offers a methodology for alleviating the problems related to the complex sensor operating relationships through message level systems integration. This methodology fits completely into all aspects of technology from existing aircraft avionics architecture, to the ATF, to the future PAVE PACE integrated architecture.

CINTEC2 was implemented as a message monitor (and message generator) that can examine messages targeted for specific avionics systems. Based upon the messages, operations were either permitted to take place, delayed, or interleaved so that avionics functionality could be preserved without destructive side effects. As a prototype, CINTEC2 is limited in functionality but clearly demonstrates the capability to functionally integrate today's existing avionics systems through mediation based upon incoming avionics systems requests, autonomous CINTEC2 avionics system usage, pilot requests, and internal/external situation awareness.

Application of the E2C2 system concepts to avionics architectures can offer a unique integration scheme which can provide enhanced capabilities in the future. Figure 26 represents the possible implementation of a CINTEC2 type system in an existing avionics system architecture for the purposes of achieving functional integration. The system would basically sit in the path of all avionics bus message traffic to and from the various aircraft systems. By monitoring the messages composed of system commands and outputs, it could effectively intercept the message, mediate and/or reconfigure the command, and output its own message to perform the task in an integrated fashion. An approach such as this can offer tremendous advantages to the overall avionics systems operations that are conducted by todays aircraft.

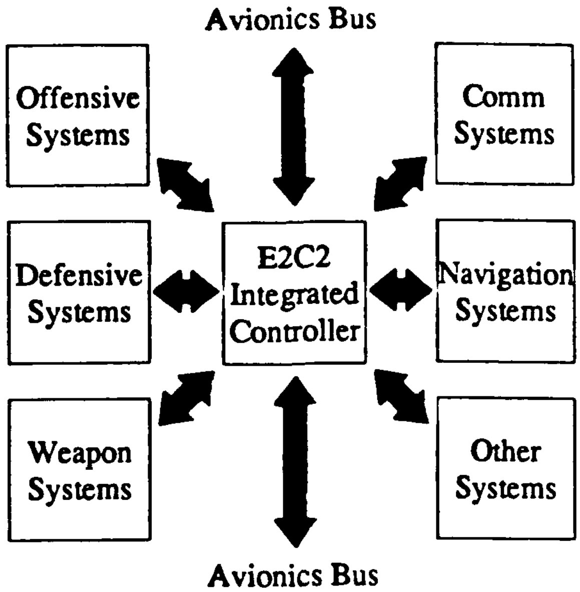


Figure 26 - CINTEC2 Avionics Systems Integration

This system concept will be especially useful since declining defense budgets will force existing aircraft to remain in the field with their current avionics systems much longer than previously thought. The approach offered by E2C2 and the CINTEC2 system design can provide an excellent backfit approach to the integration of existing avionics systems. This can be accomplished by providing the methodology for adding a single system to the existing avionics systems architectures that can perform the much needed functional integration.

Investigation into more offensive and defensive systems, as well as the remaining areas related to navigation and communication will need to be brought into the picture in order to realize the complete integration picture. However, now that a prototype system has already been developed, further refinements and research into these areas can be performed much more efficiently and economically. Conflicting or disrupted communications, effects of communications jamming, and/or navigation functions that conflict with spontaneous mission events can all be dealt with in the current CINTEC2 control philosophy. In addition, the communications aspects offer another source of information from which CINTEC2 can make decisions relating to sensor and system control.

Once all aspects of the mission, aircraft, sensor, threat, and environment have been successfully incorporated into the prototype, it will become possible to develop alternative and/or backup mediation and control strategies for system control. The goal will be to guarantee that no task, regardless of its importance, will be denied resources needed to carry out its job

completely and in a timely manner. Note, however, that in order to generate backup strategies the selection of the sensor and subsystem suite of the aircraft is very important in regard to the generation of an effective decision strategy. For this reason it is important to undertake the consideration of the attributes of the actual fielded and/or upcoming systems in continuing efforts on the prototype. This only has to be realized at the level of mode control that can currently be exercised on them (recommendations for additional control of these avionics systems can be made based upon the findings and control possibilities explored during this future effort).

Another area that would require more work is that of threat assessment. This area represents one avenue that could add a great deal of improvement to CINTEC2's ability to successfully integrate avionics systems based on external influences. By having the resources available to identify and rank various threats it would become possible to dynamically change control schemes based upon the characteristics of a given threat system. This would allow sensors and systems to be controlled and utilized in such a way as to remain more covert to a given system by using or changing frequencies of operation or knowing that low altitude effects clutter a given ground based radar. Information of this type can be exploited in a number of different ways. A natural tie-in for introducing threat information to CINTEC2 would be through the FATIL program.

Man in the loop testing is another area that needs to be explored relative to sensor mediation. The effects that the pilot could have on a system such as CINTEC2, as well as the effects that the system could have on the pilot are important aspects of overall operation. This testing would add to the value of the controller algorithms by providing valuable feedback as to the effectiveness of different sensor and system control schema while providing another baseline with which to measure CINTEC2's performance. In terms of man in the loop testing, it would be ideal to test in an environment as close to the actual avionics architecture as possible in order to completely assess system and pilot interactions. The Integrated Test Bed Facility could provide an excellent proving ground for such testing.

The flexibility to deal with all of the previously mentioned types of issues in regard to future research efforts have all been designed into the CINTEC2 software from the ground up. From the very beginning it was realized that the scope of the E2C2 program could get extremely large when trying to deal with so many different systems and so many different pieces of information. The scope of this project was limited therefore to the investigation of a few offensive and defensive avionics systems with some of the myriad external factors thrown in.

The E2C2 program designed CINTEC2 with the flexibility to incorporate additional functions in the future regardless of their specific nature. This is one of the advantages of using the expert system based prototype. Because of the need to expand, the implementation of CINTEC2 included different interfaces and software routines to handle the data structures necessary to permit easy growth of CINTEC2 into additional areas related to avionics systems. Overall, CINTEC2 and the research performed by the E2C2 program have offered an excellent avenue with which to integrate avionics systems regardless of the constraints imposed by current or future avionics system architectures. The system takes advantage of the nature of the communications that take place between avionics systems in order to perform its functions and offers a degree of hardware independence that can expand into all realms of avionics functionality. Work needs to continue, however, in order to completely realize the potential advantages of the integrated system architecture.

Appendix A -- Test Data

The following data represents the output of the CINTEC2 combat controller during its scenario test run. Simulation time precedes the actions that CINTEC2 permitted. Particular emission strategy settings and events of interest also appear in the data. The FLIR nav. update messages seen in the file indicate an upcoming waypoint crossing where the FLIR was turned on. This data was used to generate the summary statistics documented in Section 3.0.

Simulation Time = 0.00000E+00
****
LWD - Turned On
MWR - Turned On
RWR - Turned On
FLIR Nav. Update
Emission Strategy - passive

Simulation Time = 3.00000E+00
****
FLIR Nav. Update Completed

Simulation Time = 5.25000E+01
****
FLIR Nav. Update

Simulation Time = 6.00000E+01
****
FLIR Nav. Update Completed
Mission Phase Change

Simulation Time = 1.06500E+02
****
Jammer - Turned On
TFR - Turned Off
MWR - Turned Off
RWR - Turned Off
Emission strategy - active

Simulation Time = 1.09500E+02
****
Jammer Burst Terminated
TFR - Turned On
Emission strategy - lpi

Simulation Time = 1.23000E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 1.24500E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 1.27500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Optical - Turned On
Emission strategy - lpi

Simulation Time = 1.32000E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 1.41000E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 1.44000E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 1.48500E+02
****
Jammer - Turned On
TFR - Turned Off
Optical - Turned Off

Emission strategy - active

Simulation Time = 1.57500E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 1.60500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 2.04000E+02
****
Mission Phase Change

Simulation Time = 2.07000E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 2.16000E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 2.19000E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 2.32500E+02
****
TFR - Turned Off
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 2.67000E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 2.86500E+02
****
TFR - Turned Off
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 2.89500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 2.94000E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 3.03000E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 3.06000E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 3.19500E+02
****
TFR - Turned Off
MWR - Turned On

RWR - Turned On
Emission strategy - passive

Simulation Time = 3.33000E+02
****
FLIR Nav. Update

Simulation Time = 3.42000E+02
****
FLIR Nav. Update Completed
Mission Phase Change

Simulation Time = 4.15500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 4.18500E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 4.20000E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 4.21500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 4.30500E+02
****
FLIR Nav. Update

Simulation Time = 4.35000E+02
****
Laser Designator - Turned On

LWD - Turned Off
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 4.36500E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 4.41000E+02
****
Laser Designator - Turned Off
LWD - Turned On
FLIR Nav. Update Completed
Mission Phase Change
Emission strategy - passive

Simulation Time = 4.48500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 4.53000E+02
****
TFR - Turned Off
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 4.56000E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 4.60500E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 4.69500E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - passive

Simulation Time = 4.72500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 4.77000E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 4.86000E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Emission strategy - p.

Simulation Time = 4.89000E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 4.93500E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 5.02500E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On

Emission strategy - passive

Simulation Time = 5.05500E+02
****
TFR - Turned On
MWR - Turned Off
RWR - Turned Off
Emission strategy - lpi

Simulation Time = 5.10000E+02
****
Jammer - Turned On
TFR - Turned Off
Emission strategy - active

Simulation Time = 5.19000E+02
****
Jammer Burst Terminated
MWR - Turned On
RWR - Turned On
Optical - Turned On
Emission strategy - passive

Simulation Time = 5.43000E+02
****
FLIR Nav. Update

Simulation Time = 5.50500E+02
****
Optical - Turned Off

*** End of Scenario ***

Appendix B -- Decision Strategy Rule Set

The following file listing represents the CINTEC2 decision strategy that was implemented through the MeriTool inferencing software. The format of the particular rules and data structures is discussed in the MeriTool 2.2 Software User's Manual.

```
/* --------------- Meritool Rule File (cb.rul) ------------------------ */
/* ------------- DEBUGGING DIRECTIVES --------------------- */

/* watch_run. shows rule firings */

/* watch_cache. shows changes to attribute values and variables */

/* uncertainty confidence. gives uncertainty confidence to variables */


/* ---------------------- FUNCTION DIRECTIVES ----------------------- */
function output_message(integer, integer).
function activate_ruleset(integer).
function deactivate_ruleset(integer).

/* --------------- DEFINITIONS OF OBJECTS AND ATTRIBUTES --------------- */

object information has
  c_time
  active1
  active2
  active3
  mission_phase_change
  terrain
  timestate
  first_time.

import information with
  c_time        float
  active1       symbol        /* active or nonactive sensor rule set */
  active2       symbol  /* active or nonactive threat rule set */
  active3       symbol  /* active or nonactive mission rule set */
  mission_phase_change symbol
  terrain       symbol
  timestate  symbol
  first_time    symbol.
```

```
object aircraft has
  lat
  long
  alt
  speed
  heading
  pitch
  roll
  yaw.

object mission has
  roe
  emiss_strat
  phase
  status
  t_lat
  t_lon
  n_wpt_lat
  n_wpt_lon
  n_wpt_spd
  n_wpt_alt
  n_wpt_time_to.

object sensor has
  sim_t
  name
  role
  mode
  ready
  g_azm
  g_ele
  g_rng
  az_res
  ele_res
  rng_res.

object snsr_conf has
  snsr_one
  snsr_two
  start_t
  end_t.

object track_f has
  sim_t
  tar_num
```

```
  tar_az
  tar_ele
  tar_rng
  tf_ffn
  tf_class
  tf_type
  tf_mode
  tf_conf.

object snsr_rpt has
  sim_t
  name
  tar_num
  tar_az
  tar_ele
  tar_rng
  sr_ffn
  sr_class
  sr_type
  sr_mode
  sr_conf.

object snsr_req has
  sim_t
  exe_t
  end_t
  req_snsr
  snsr_mode
  snsr_az
  snsr_ele
  snsr_rng
  pri
  status.

object weapon has
  name
  m_c_guid
  m_c_snsr
  ter_guid
  ter_snsr
  num_remain
  num_in_flight
  status.

/* import the needed objects */
```

```
import aircraft with
  lat          float
  long         float
  alt          float
  speed        float
  heading      float
  pitch float
  roll         float
  yaw          float.

import mission with
  roe          symbol
  emiss_strat  symbol
  phase        symbol
  status       symbol
  t_lat  float
  t_lon float
  n_wpt_lat    float
  n_wpt_lon    float
  n_wpt_spd    float
  n_wpt_alt    float
  n_wpt_time_to float.

import sensor with
  sim_t        float
  name         symbol
  role         symbol
  mode                     symbol
  ready        symbol
  g_azm        symbol
  g_elesymbol
  g_rng        symbol
  az_res       float
  ele_res      float
  rng_res      float.

import snsr_conf with
  snsr_one     symbol
  snsr_two     symbol
  start_t      float
  end_t        float.

import track_f with
  sim_t        float
  tar_num      integer
  tar_az       float
```

```
   tar_ele      float
   tar_rng      float
   tf_ffn       symbol
   tf_class     symbol
   tf_type      symbol
   tf_mode      symbol
   tf_conf      float.

import snsr_rpt with
   sim_t        float
   name         symbol
   tar_num      integer
   tar_az       float
   tar_ele      float
   tar_rng      float
   sr_ffn       symbol
   sr_class     symbol
   sr_type      symbol
   sr_mode      symbol
   sr_conf      float.

import snsr_req with
   sim_t        float
   exe_tfloat
   end_t        float
   req_snsr     symbol
   snsr_mode    symbol
   snsr_az      float
   snsr_ele     float
   snsr_rng     float
   pri          integer
   status       symbol.

import weapon with
   name                 symbol
   m_c_guid             symbol
   m_c_snsr             symbol
   ter_guid             symbol
   ter_snsr             symbol
   num_remain           integer
   num_in_flight        integer
   status               symbol.

/******************/
/* Invoke the rules */
/*****************/
```

```
/* ---------------------- The Mission Ruleset ----------------------- */

mr1: if sensor with name = mmr and
                          mode = on and
        mission ?m1 with emiss_strat <> active
   then
        output_message(3,1)
        modify(?m1 with emiss_strat = active).

mr2: if sensor with name = jammer and
                          mode = on and
        mission ?m1 with emiss_strat <> active
   then
        output_message(3,1)
        modify(?m1 with emiss_strat = active).

mr3: if sensor with name = chaff and
                          mode = on and
        mission ?m1 with emiss_strat <> active
   then
        output_message(3,1)
        modify(?m1 with emiss_strat = active).

mr4: if sensor with name = flare and
                          mode = on and
        mission ?m1 with emiss_strat <> active
   then
        output_message(3,1)
        modify(?m1 with emiss_strat = active).

mr5: if sensor with name = tfr and
                          mode = on and
        mission ?m1 with emiss_strat <> lpi and
                           phase <> target_attack
   then
        output_message(3,2)
        modify(?m1 with emiss_strat = lpi).

mr6: if sensor with name = laser and
                          mode = on and
        mission ?m1 with emiss_strat <> lpi and
                           phase <> target_attack
   then
        output_message(3,2)
        modify(?m1 with emiss_strat = lpi).
```

```
mr7: if mission ?m1 with emiss_strat = active and
                         roe <> weapons_free
   then
       output_message(3,3)
       modify(?m1 with roe = weapons_free).

mr8: if mission ?m1 with emiss_strat = lpi and
                         roe <> weapons_free
   then
       output_message(3,4)
       modify(?m1 with roe = weapons_free).

mr9: if mission ?m1 with emiss_strat = passive and
                         roe <> weapons_tight
   then
       output_message(3,4)
       modify(?m1 with roe = weapons_tight).

mr10: if sensor with name = rwr and
                         mode = on and
       mission ?s1 with emiss_strat <> passive and
                          phase <> target_attack
    then
       output_message(3,6)
       modify(?s1 with emiss_strat = passive).

/* ---------------------- The Sensor Ruleset ----------------------- */

/* ----- passive sensor control -------- */

sr1: if sensor with name = jammer and
                         mode = off and
       sensor with name = tfr and
                         mode = off and
       sensor with name = mmr and
                         mode = off and
       sensor ?s1 with name = rwr and
                         mode = off
   then
       output_message(1,1)
       modify(?s1 with mode = on).

sr2: if sensor with name = jammer and
                         mode = off and
       sensor with name = tfr and
                         mode = off and
```

```
        sensor with name = mmr and
                          mode = off and
        sensor ?s1 with name = mwr and
                          mode = off
    then
        output_message(1,2)
        modify(?s1 with mode = on).

sr3: if sensor ?s1 with name = lwd and
                          mode = off and
         sensor with name = laser and
                          mode = off
    then
        output_message(1,3)
        modify(?s1 with mode = on).

sr4: if sensor with name = laser and
                          mode = on and
        sensor ?s1 with name = lwd and
                          mode = on
    then
        output_message(1,18)
        modify(?s1 with mode = off).

sr5: if sensor with name = jammer and
                          mode = on and
        sensor ?s1 with name = rwr and
                          mode = on
    then
        output_message(1,19)
        modify(?s1 with mode = off).

sr6: if sensor with name = mmr and
                          mode = on and
        sensor ?s1 with name = rwr and
                          mode = on
    then
        output_message(1,19)
        modify(?s1 with mode = off).

sr7: if sensor with name = tfr and
                          mode = on and
        sensor ?s1 with name = rwr and
                          mode = on
    then
        output_message(1,19)
```

```
        modify(?s1 with mode = off).

sr8: if sensor with name = jammer and
                         mode = on and
        sensor ?s1 with name = mwr and
                         mode = on
   then
        output_message(1,20)
        modify(?s1 with mode = off).

sr9: if sensor with name = mmr and
                         mode = on and
        sensor ?s1 with name = mwr and
                         mode = on
   then
        output_message(1,20)
        modify(?s1 with mode = off).

sr10: if sensor with name = tfr and
                         mode = on and
        sensor ?s1 with name = mwr and
                         mode = on
   then
        output_message(1,20)
        modify(?s1 with mode = off).

/* ----------- end passive sensor control rules -------- */

sr17: if sensor ?s1 with name = raltimeter and
                         mode = off and
        sensor with name = tfr and
                    mode = on and
        information with timestate <> ten
   then
        modify(?s1 with mode = on)
        output_message(1,5).

sr18: if track_f with tf_mode = track and
        sensor with name = tfr and
                       mode = on and
        sensor ?s1 with name = jammer and
                       mode = off and
        information with timestate = five
   then
        output_message(1,8)
        modify(?s1 with mode = on).
```

```
sr19: if track_f with tf_mode = track and
         sensor ?s1 with name = tfr and
                         mode = on and
         sensor  with name = jammer and
                         mode = on
    then
         output_message(1,6)
         modify(?s1 with mode = off).

sr20: if track_f with tf_mode = track and
         sensor with name = tfr and
                         mode = off and
         sensor ?s1 with name = jammer and
                         mode = on and
         information with timestate = one
    then
         output_message(1,24)
         modify(?s1 with mode = off).

sr21: if track_f with tf_mode = track and
         sensor ?s1 with name = tfr and
                         mode = off and
         sensor  with name = jammer and
                         mode = off and
         information with timestate <> ten
    then
         output_message(1,4)
         modify(?s1 with mode = on).

sr22: if sensor ?s1 with name = raltimeter and
                            mode = on and
         sensor with name = tfr and
                            mode = off
    then
         output_message(1,7)
         modify(?s1 with mode = off).

sr23: if snsr_rpt with name = lwd and
         sensor ?s1 with name = chaff and
                     mode = off
    then
         output_message(1,10)
         modify(?s1 with mode = on).

sr24: if snsr_rpt with name = mwr and
```

```
        sensor ?s1 with name = chaff and
                      mode = off
    then
        output_message(1,10)
        modify(?s1 with mode = on).

sr25: if snsr_rpt with name = lwd and
        sensor ?s1 with name = flares and
                      mode = off
    then
        output_message(1,11)
        modify(?s1 with mode = on).

sr26: if snsr_rpt with name = mwr and
        sensor ?s1 with name = flares and
                      mode = off
    then
        output_message(1,11)
        modify(?s1 with mode = on).


sr27: if mission with n_wpt_time_to < 10.0 and
        sensor ?sen with name = flir and
                        mode = off
    then
        output_message(1,12)
        modify(?sen with mode = on).

sr28: if mission with n_wpt_time_to > 10.0 and
        sensor ?var with name = flir and
                        mode = on
    then
        output_message(1,13)
        modify(?var with mode = off).

sr29: if track_f with tar_az > 30.0 and
                        tar_az < 330.0 and
        sensor ?s1 with name = optical and
                        mode = off and
        mission with roe = weapons_free
    then
        output_message(1,14)
        modify(?s1 with mode = on).

sr30: if mission with roe = weapons_tight and
        sensor ?s1 with name = optical and
```

```
                        mode = on
    then
       output_message(1,15)
        modify(?s1 with mode = off).

sr31: if mission with roe = weapons_tight and
          sensor ?s1 with name = laser and
                        mode = on
    then
       output_message(1,17)
        modify(?s1 with mode = off).

sr32: if mission with n_wpt_time_to < 5.0 and
                    phase = target_attack and
         sensor ?s1 with name = laser and
                        mode = off
   then
       modify(?s1 with mode = on)
       output_message(1,16).

sr33: if mission with n_wpt_time_to > 5.0 and
         sensor ?s1 with name = laser and
                        mode = on
   then
       modify(?s1 with mode = off)
       output_message(1,17).

sr34: if information with timestate = ten and
           sensor ?s1 with name = tfr and
                        mode = on
    then
       modify(?s1 with mode = off).

sr35: if information with timestate = ten and
           sensor ?s1 with name = jammer and
                        mode = on
    then
       modify(?s1 with mode = off).

sr36: if information with timestate = ten and
           sensor ?s1 with name = raltimeter and
                        mode = on
    then
       modify(?s1 with mode = off).

/* ---------------------- Environment Ruleset ----------------------- */
```

```
/* ---------- rules based on prebriefed environment information -------- */

er1: if mission with phase = cruise and
        information ?info with terrain <> smooth
   then
        modify(?info with terrain = smooth).

er2: if mission with phase = ingress and
        information ?info with terrain <> smooth
   then
        modify(?info with terrain = smooth).

er3: if mission with phase = target_acquisition and
        information ?info with terrain <> rolling
   then
        modify(?info with terrain = rolling).

er4: if mission with phase = target_attack and
        information ?info with terrain <> rugged
   then
        modify(?info with terrain = rugged).

er5: if mission with phase = egress and
        information ?info with terrain <> rugged
   then
        modify(?info with terrain = rugged).

/* ---------------------- Generic Ruleset/Initialization -------------- */

r1: if information ?info with c_time = 0.1 and
                              first_time = yes
   then
         modify(?info with active1 = yes and
                           active2 = no and
                           active3 = no and
                           first_time = no).

r2: if information ?info with mission_phase_change = yes
   then
        modify(?info with active3 = yes and
                          mission_phase_change = no)
        output_message(3,5).

r3: if mission ?m1 with phase = target_attack and
                        emiss_strat <> active
   then
```

```
output_message(3,1)
modify(?m1 with emiss_strat = active).
```


☆U.S. GOVERNMENT PRINTING OFFICE: 1992 - 648-127/62576